THE UNIVERSITY *of* EDINBURGH

Curvilinear Shallow Flow and Particle Tracking Model for a Groyned River Bend

Mohammad Mahdi Jalali

A thesis submitted for the degree of

Doctor of Philosophy

The University of Edinburgh

2016

Abstract

Hydraulic structures such as dykes and groynes are commonly used to help control river flows and reduce flood risk. The present research aims to develop an idealized model of the hydrodynamics in the vicinity of a large river bend, and the advection and mixing processes where groynes are located. In this study a curvilinear model of shallow water equations is applied to investigate chaotic advection of particles in a river bend similar in dimensions to a typical bend in the River Danube, Hungary. First, a curvilinear grid generator is developed based on Poisson-type elliptic partial differential equations. The grid generator is verified for benchmark tests concerning a circular domain and for distorted grids in a rectangular domain. It is found that multi-grid (MG) and conjugate gradient (CG) methods performed better computationally than successive over-relaxation (SOR) in generating the curvilinear grids. The open channel hydrodynamics are modelled using the shallow water equations (SWEs) derived by depth-averaging the continuity and Navier-Stokes momentum equations. Both Cartesian and curvilinear forms of the shallow water equations are presented. Both sets of equations are discretized spatially using finite differences and the solution marched forward in time using fourth-order Runge-Kutta scheme. The shallow water solvers are verified and validated for uniform flow in the rectangular channel, wind-induced set up in rectangular and circular basins, flow past a sidewall expansion, and Shallow flow in a rectangular channel with single groyne. A Lagrangian particle tracking model is used to predict the trajectories of tracer particles, and bilinear interpolation is used to provide a representation of the continuous flow field from discrete results. The particle tracking model is verified for trajectories in the flow field of a single free vortex and in the alternating flow field of a pair of blinking vortices. Excellent agreement is obtained with analytical solutions, previously published results in the literature. The combined shallow flow and Lagrangian particle tracking model is then used to simulate particle advection in the flow past a side-wall cavity containing a groyne and reasonable agreement is obtained with published experimental and alternative numerical data. Finally, the combined model is applied to simulate the shallow flow hydrodynamics, advection and mixing processes in the vicinity of groynes in river bend, the dimensions representative of a typical bend in the Danube River, Hungary.

Acknowledgments

I have to give my heartfelt thanks to my dear supervisor professor Alistair Borthwick for accepting me as his PhD student and giving me the opportunity to spend the best time of my life in Edinburgh University, U.K and I appreciate his ongoing support. Such a fantastic experience and incredible achievement would not be possible without his kind assistance and advice. I am really grateful to this great man who gave me confidence as well as his unique knowledge to fulfil my accomplishment. He will live in my heart for ever. I also want to dedicate this PhD thesis to my kind mother who brought happiness to my heart when the hardships made me miserable telling me that I could overcome all difficulties if I believed in myself. I want to dedicate my work to my father who taught me to be strong and never give up in my life. I know I can never ever repay their sacrifices and kindness. I want to thank my twin brother who shared with me all the difficult as well as the joyful moments during my studies. He showed me the wrong and right in life and gave me the choice to make my own decision. Last but not least, I want to thank my Advisor Dr. Vengatesan Venugopal for his kind support. I am also thankful to Professor Hugh McCann and School of Engineering in University of Edinburgh for providing me with great conditions to finish my PhD studies in the best way.

Declaration

I confirm that the present thesis is based on my PhD researches under the supervision of Professor Alistair Borthwick. All the simulations and results are obtained by means of the developed numerical and analytical models. This thesis is prepared for degree of Doctor of Philosophy from University of Edinburgh.

Mohammad Mahdi Jalali 27 July 2016

Table of Contents

Nomenclature

X	Horizontal axis in Cartesian system
Y	Vertical axis in Cartesian system
ξ	Horizontal axis in transformed system
η	Vertical axis in transformed system
S	Boundary of physical domain in Cartesian system
T	Boundary of transformed domain in transformed system
P	Expressions for concentrating ξ – lines
Q	Expressions for concentrating η – lines
J	The transformation Jacobian
ω	An acceleration parameter
u	The exact solution for Multi-grid method
v	An approximate solution for Multi-grid method
k	The matrix in Multi-grid and conjugate gradient methods
e	The error for Multi-grid and conjugate gradient methods
r	The residual for Multi-grid and conjugate gradient methods
q_0	The residual for conjugate gradient methods
a	The amplitude
d	Decay factor
Δx	The elemental dimensions in *x*-direction
Δy	The elemental dimensions in *y*-direction
h	Total depth of water

h_s	Still water depth
ζ	Free surface elevation above still water level
$\bar{u}$	Velocity in *x*-direction
$\bar{v}$	Velocity in *y*-direction
$\bar{w}$	Velocity in *z*-direction
ρ	Water density
P	Pressure
z_b	Elevation of the bottom surface boundary above a fixed horizontal datum
g	Gravity
Δt	Time stepping interval
q_x	Discharge in *x*-direction
q_y	Discharge in *y*-direction
τ_{bx}	Bed friction in *x*-direction
τ_{by}	Bed friction in *y*-direction
τ_{wx}	Wind resistance in *x*-direction
τ_{wy}	Wind resistance in *y*-direction
ω	Angular speed of earth
$\emptyset$	Latitude speed of earth
i_{max} and j_{max}	Number of grid in *x*- direction and *y*-direction
C	Chézy coefficient
D	The diameter of circular basin
S	Bed Slop

b	Width of channel with side wall expansion
b_c	Width of channel at the centre of the velocity distribution element
R_{eI}	Reynolds number
U_I	Velocity at the inlet
υ	Eddy viscosity
r_0	The interior radius of Danube bend
r_1	The exterior radius of Danube bend
θ	Angular axis in polar system
u_r	Radial velocity component
u_θ	Tangential velocity component
ψ	Free vortex function
Γ	Strengths of the vortex
i_p and j_p	The specific cell position for a particle
x_p	Particle position in *x*- direction
y_p	Particle position in *y*- direction
u_p	Particle velocity in *x*- direction
v_p	Particle velocity in y- direction

Chapter 1

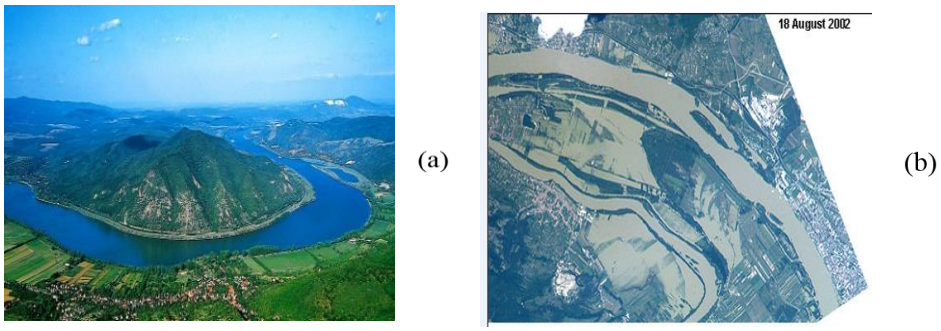
Introduction and Literature Review

Chapter 1 presents an introduction to the use of groynes in rivers, mixing processes in large rivers, and a review of related literature. The chapter then lists the aim and objectives of this thesis on shallow flow and particle advection in a large river bend containing groynes. The final section outlines the overall structure of the thesis.

1.1 Groynes in Large Rivers

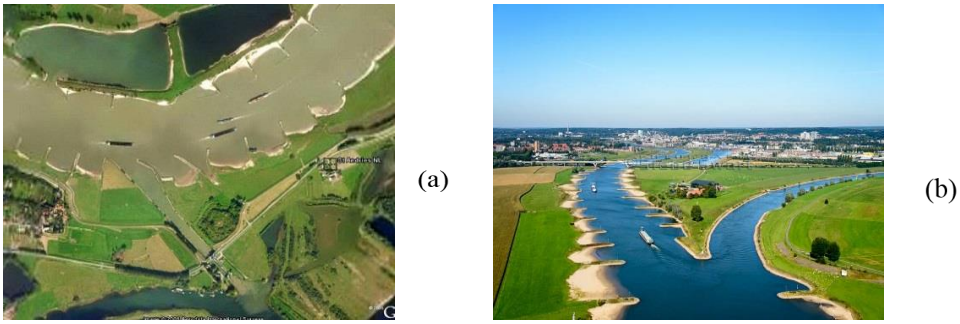
A groyne is a solid hydraulic structure that projects out of a riverbank to interrupt sediment transport. Groynes are constructed along wide rivers to supplement flood protection works, and act partly to slow the river flow, and partly to accumulate sediment. The presence of curved lateral boundaries and morphological features in an untrained river has a significant influence on river flow, and may reduce the flood capacity at river bend especially when extreme rainfall and snowmelt events occur. Throughout recorded history river flooding has caused severe economic damage, even ruining national economies, and very large numbers of fatalities (even exceeding one million in a single event involving the Yellow River, China). A secondary phenomenon caused by the presence of river bends is the high level of sedimentation that can occur due to depositional processes as a flood subsides. The transport of river sediments by floods onto nearby land can enormously disrupt rural and agricultural land-use close to rivers, and greatly impact on the lives of nearby residents. The cost of sediment damage is therefore of great importance to flood risk management. One way of mitigating against flood risk in large rivers is by constructing groynes to guide and collect sediments inside the river. Although groynes have advantages regarding reduced flood risk, they are expensive to construct and maintain, and can affect navigation because of flow separation at the ends of groynes and the production of turbulent eddies and larger-scale recirculation zones. Noting the abovementioned hydraulic problems, it is a challenge for engineers to design groynes that control the river meeting flood risk, navigation and economic cost objectives. A particular case of

interest is the Danube river, the second largest in Europe. The Hungarian part of Danube is characterized by a series of bends, a typical example shown in Figure 1.1a before a flood event. Figure 1.1b shows the high levels of sediment loading in the environs of Budapest after a major flood in 2002. The photographs were taken by NASA Earth Observatory[1]. Figure 1.2 shows typical rivers bends that have been trained by groynes against extreme flood and sediment events.



(a) (b)

Figure 1.1 Danube River, Hungary: (a) Danube River bend, Budapest immediately before the 2002 flood, obtained from (http://www.minibusbudapest.com/danube-bend-tour), and (b) Danube River bend, Budapest, Hungary after 2002 flood, obtained from (http://www.earthobservatory.nasa.gov/IOTD/view.php?id=2743)



(a) (b)

Figure 1.2 Groynes in Waal River bends, Netherlands: (a) Sediment-laden river, obtained from (albionrose.com/id81.htm), and (b) Clear river, obtained from (http://www.flickrhivemind.net/Tags/controlworks/Interesting/)

The Yellow River (Huang He River) is the second longest river in China, and the most heavily sediment-laden. Each year there is a spring flood flow in the Yellow River due to heavy rainfall, which is exacerbated by blocking of the river bends caused

[1] http://www.earthobservatory.nasa.gov

by slow ice melting. Based on NASA earth observatory data report the northern Yellow River bends pose the worst flood problems for river managers[1]. These 180^{o} bends become blocked by the ice at the apex of each bend, forcing the mainstream to overflow the riverbanks and inundate rural areas. Figure 1.3 depicts the Yellow River flood in March 2008, when more than 13,000 people had to be rescued, in areas close to the northern bends and considerable economic damage was sustained.

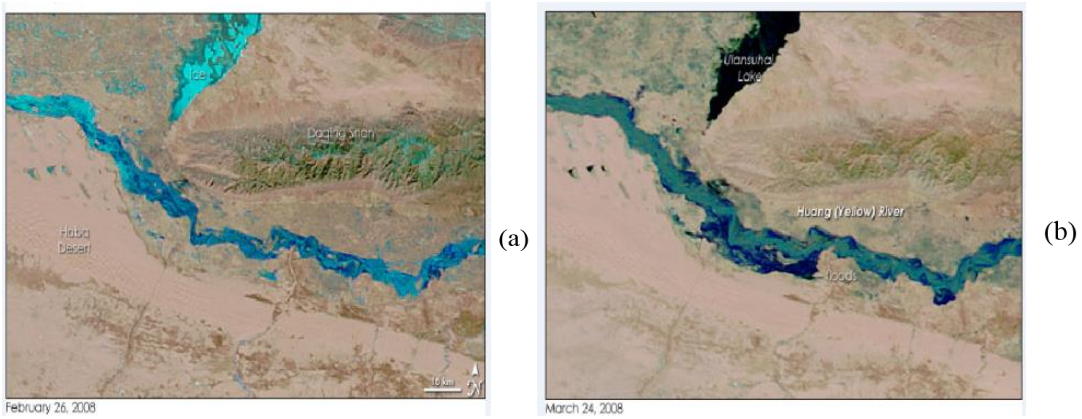


Figure 1.3 Yellow River flood in 2008, China: (a) before the flood, 26 February, (b) after the flood, 24 March. obtained from (http://www.earthobservatory.nasa.gov/NaturalHazards/view.php?id=19723)

1.2 Mixing Processes in Large Rivers

Mixing occurs due to advection and dispersion processes (see e.g. Rutherford, 1981) whereby species become distributed within a flow field. Mixing processes are promoted by shear and vorticity in a shallow flow, primarily through turbulent eddies, and have a direct influence on transport of material, heat, contaminants, etc. and thus affect water quality. Advection is where particles move with locally time-averaged flow velocity components, and depends on flow direction. Dispersion involves the separation of particles in a river, and is driven by diffusion and non-uniformities in velocity. Mixing can occur in both laminar and turbulent flows. Molecular diffusion takes place at a much slower rate (corresponding to the kinematic viscosity coefficient

[1] http://www.earthobservatory.nasa.gov

of the fluid, which is about 1 x 10^{-6} m^2/s) than flow fluctuations due to turbulence (which may correspond to an eddy viscosity in the order of 1-100 m^2/s). In laminar flows, diffusion is entirely due to molecular processes because of the absence of turbulence. In turbulent flows, the eddy viscosity dominates and so the turbulent velocity gradient drives dispersion in the mixing process. In wide rivers, turbulent diffusion depends on depth, bed stress, and shear velocity. An important factor in river mixing is the presence of large-scale eddies, which may be created from the bed boundary layer as it rolls up into eddies and from flow separation at obstacles and the side wall boundary layers; such eddies play a major role in entraining sediment into suspension and driving sediment transport in a river. Kolk-boils are large-scale vortical structures produced over a duned bed that also affect mixing. Dispersion tends to reduce considerably in the viscous sub-layer of the bed boundary layer, because the velocity is much slower and conditions are laminar in this near-bed region. In terms of suspended sediment, the mixing of coarse sediment particles affected by large eddies is more visible compared to that of finer particle sizes (Socolofsky and Jirka, 2005). It is usual to divide mixing in a river into three main categories (see e.g. Rutherford, 1981). The first type of mixing in the river is vertical mixing where turbulence is induced by velocity shear, and the dispersion process depends on depth and shear velocity. The second type of river mixing known as transverse mixing depends on the turbulence intensity, the presence of transverse secondary currents, and the river width. The turbulence scale for transverse mixing is based on depth so that larger eddies cause more dispersion in the transverse direction compared to the vertical direction. Transverse mixing therefore depends on both depth and shear velocity. The third type of mixing in a river is longitudinal dispersion controlled by non-uniform velocity and advection zones. In longitudinal dispersion, the mixing process involves is separation of particles in the stream-wise direction. In this case, sediment transport occurs at a faster speed in the middle of channel compared to the bed and riverbanks. The three mixing processes (vertical, transverse and longitudinal mixing) are usually studied and measured by tracer separation procedure, turbulent diffusion, dispersion rate, and velocity distribution (Day and Wood, 1976), (Deng, *et al*. 2001). Figure 1.4 is a definition sketch depicting the velocity gradient, and dispersion rate (reproduced from

Rutherford, 1981). Figure 1.5 shows the longitudinal mixing in the Waikato River measured by Rutherford *et al*. (1980).

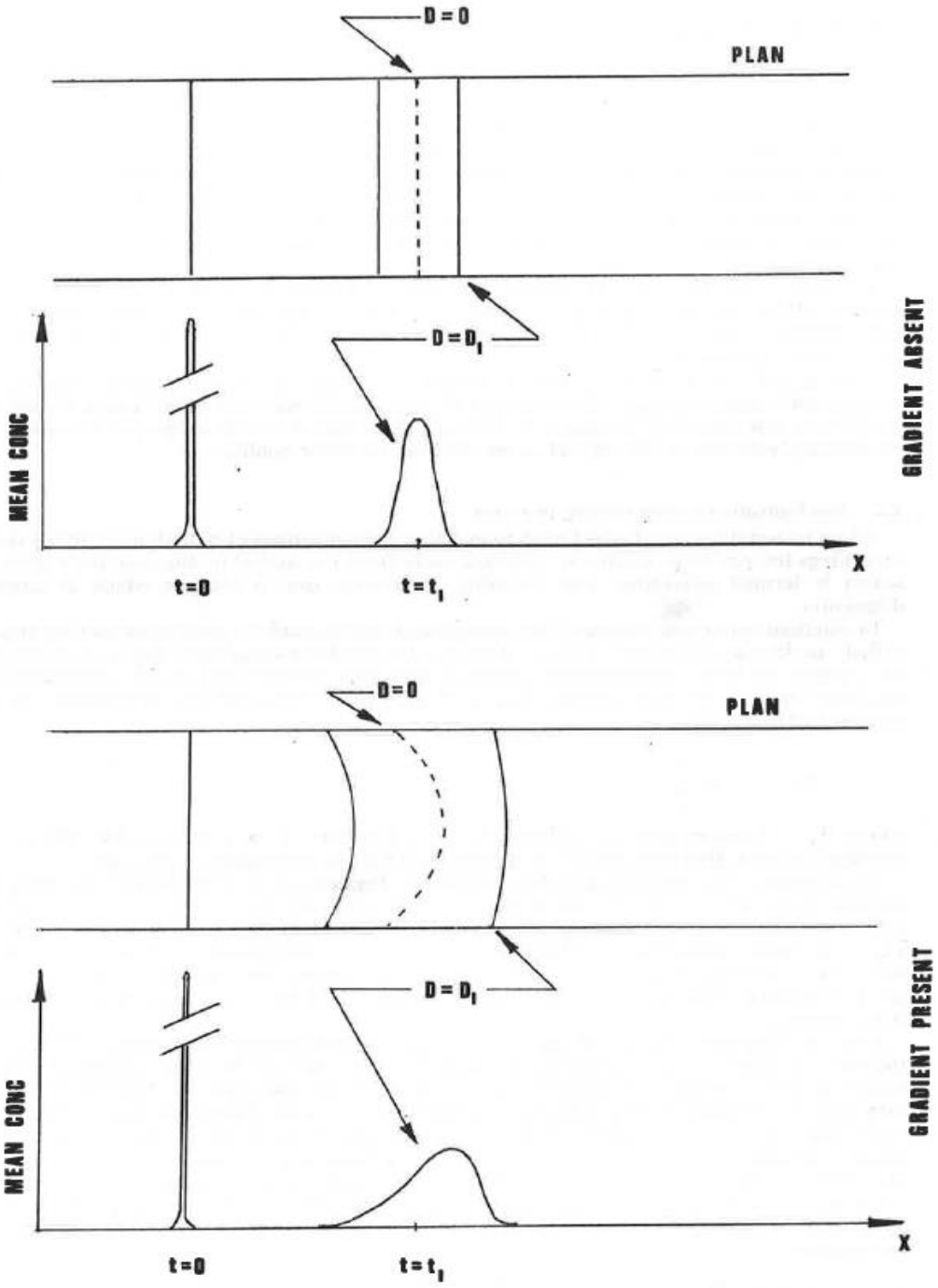


Figure 1.2 Sketch showing how a velocity gradient increases the dispersion rate.

Figure 1.4 Definition sketch showing velocity gradient and rate of dispersion rate of dispersion, reproduced from Rutherford (1981)

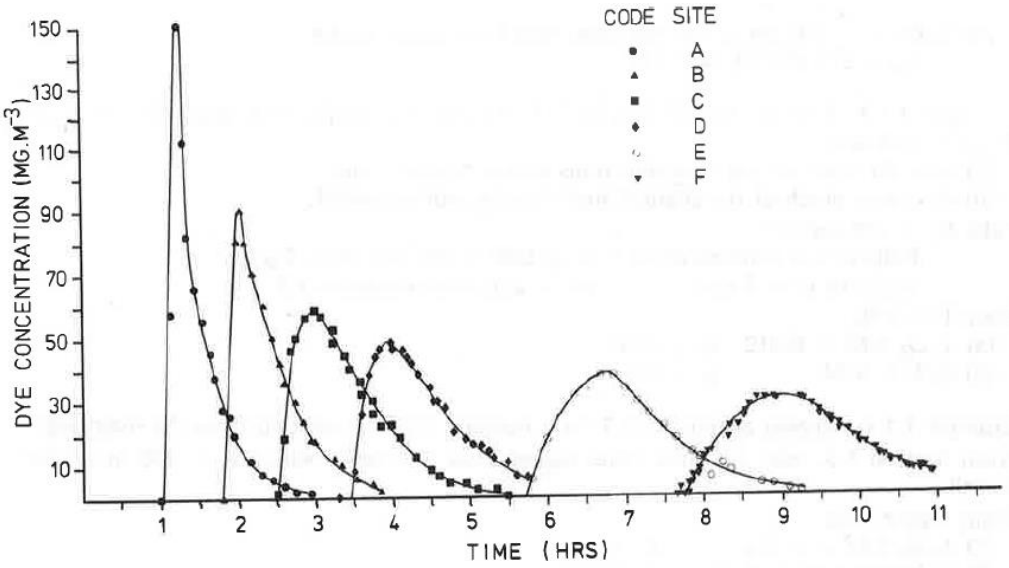


Figure 4.1 Longitudinal dispersion of dye in the Waikato River (after Rutherford *et al.* 1980).

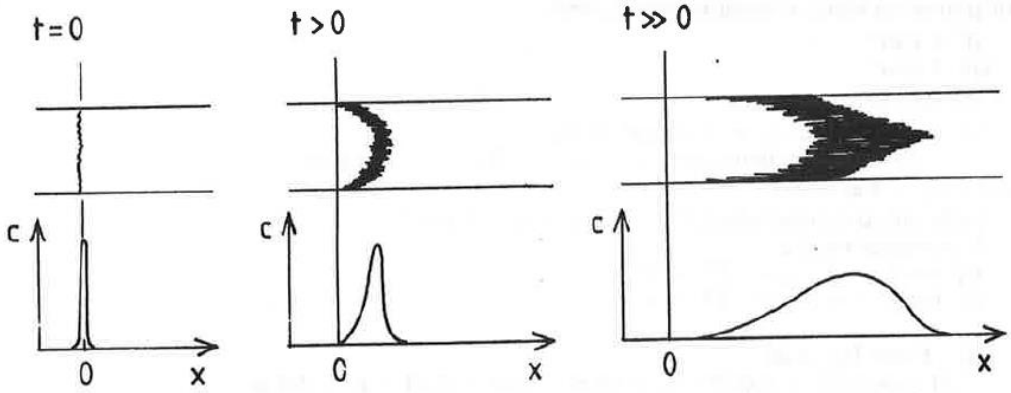


Figure 4.2 The effect of transverse velocity gradients and dispersion on longitudinal dispersion.

Figure 1.5 Longitudinal mixing definition sketches based on velocity gradient, dispersion and diffusion rate in the Waikato River, New Zealand, reproduced from Rutherford (1981)

Fluvial processes are caused by natural changes to the river shape and its flow-bed interaction; however, the river flow regime, which may be the upper, middle or lower reaches, influences erosion and fluvial mixing in a river (Day, 2014). The most important factor in fluvial mixing is the local river pattern. Perhaps the most interesting pattern is the meandering river, which includes different curved boundaries and bends that simultaneously affecting the flow hydrodynamics and mixing processes. In a meandering river, flow near the free surface migrates toward the outer bank whereas the near-bed flow migrates toward the inner bank, which causes erosion to occur at the outer bank. This process repeats itself bend by bend along the river (see Figure 1.6).

Figure 1.6 Meandering pattern in the river Morava, Czech Republic (photograph obtained from https://www.flickr.com/photos/hrubos/21360264055)

Advection and dispersion of particles can often be easily discerned when two different rivers combine and merge with each other. The photographic images in Figure 1.7 show the interaction between a sediment-laden river and a relatively clear river in Brazil and in China. In both cases, mixing processes take place over considerable longitudinal distances down the river, and sharp fronts are evident as the river branches merge.

(a)

(b)

Figure 1.7 Sediment particle mixing at the junction between sediment laden and pure water rivers: (a) Negro and Solimões Rivers in the Brazilian jungle (http://rai77.ru), (b) Jialing and Yangtze rivers in China (http://twistedsifter.com/2012/04/confluences-around-the-world/)

NASA's Earth Observatory data[1] confirm the importance of the Yellow River, China, as one of the world's most sediment-laden rivers. The source of the Yellow River is on the Qinhai-Tibetan Plateau, and most of the sediment enters the middle Yellow River from erosion of the Loess Plateau. Much of the sediment transported along the Yellow River enters the Bohai Sea via the Yellow River Delta, which is itself shaped by morphological changes connected with the sediment transport processes in the Yellow River. Since the establishment of the People's Republic of China in 1949, many flood countermeasures and water-sediment conservation measures have been implemented along the Yellow River; notably, sediment check dams along the middle Yellow River, large river control dams (such as at Xiaolangdi) and renewed dykes along the lower Yellow River. Many groynes have also been introduced to train the river. Figure 1.8 shows the substantial changes that have occurred to the Yellow river delta from 1995 to 2009. The changes are primarily related to manmade constructions, including a new channel that was initiated in the late 1990s on the north side of the delta. The cumulative effect of all the sediment conservation work has been to cause the Yellow River to become rather less yellow in recent years because the sediment concentration has fallen in the river at a faster rate than the reduction in flow discharge.

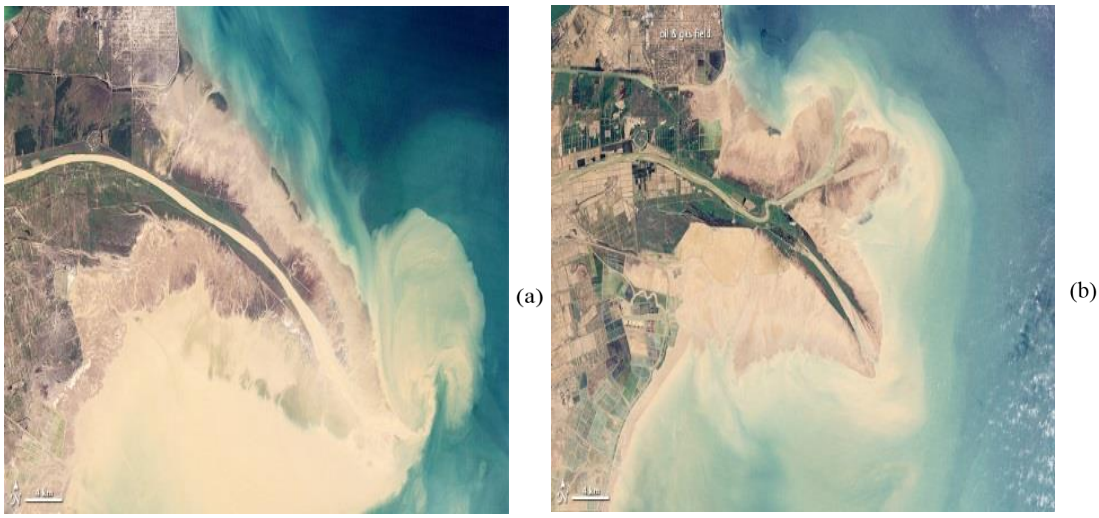


Figure 1.8 Yellow River delta morphology and sediment distributions, China: (a) 18 Sep 1995, (b) 20 June 2009, obtained from (http://earthobservatory.nasa.gov/Features/WorldOfChange/yellow_river.php)

[1] http://www.earthobservatory.nasa.gov

The Mississippi River also carries a very large sediment load, and carries 500 million tons of sediment per annum into the Mexico Gulf [1]. Figure 1.9a shows an aerial view of the Mississippi River as it enters the Gulf of Mexico. State government agencies are responsible for managing the river and protecting the coastline against sediment accretion and/or erosion; there is an annual budget for the maintenance and building of new groynes and levees along the Mississippi River[2]. In Europe, the Guadalquivir River, Spain transports sediment into the Gulf of Cádiz. Figure 1.9b is also taken from the NASA Earth Observatory and shows a satellite image of the sediment discharge into the Gulf of Cádiz in November 2012. It is evident that the mixing processes in the Gulf of Cádiz[3] are dominated by advection of nearshore currents, with the sediment plume beginning to roll up into a whorl.

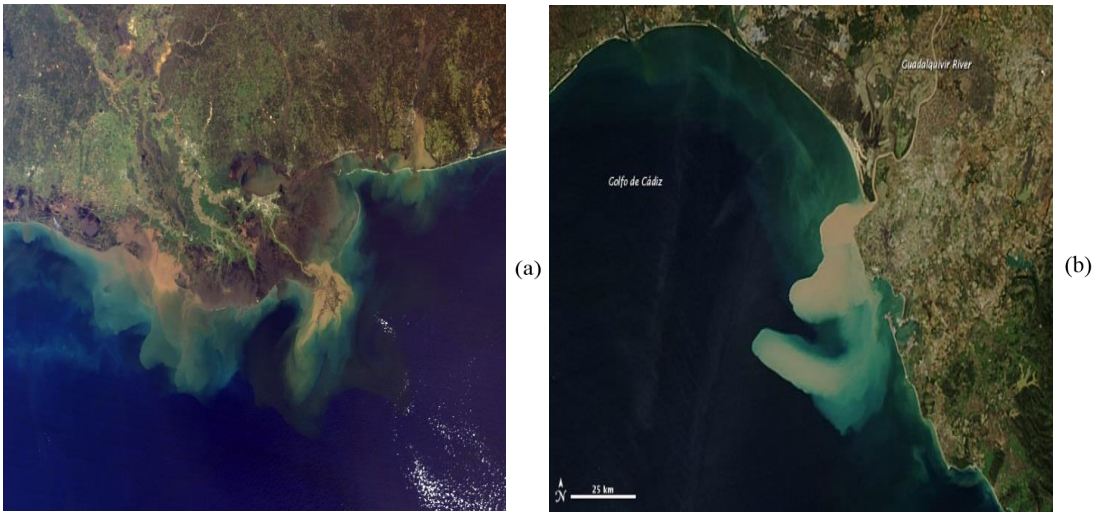


Figure 1.9 Satellite views of the Mississippi River in US and the Guadalquivir River in Spain : (a) Mississippi River as it enters the Gulf of Mexico, obtained from (http://earthobservatory.nasa.gov/IOTD/view.php?id=1257), (b) The Guadalquivir river sediment fields, obtained from (http://earthobservatory.nasa.gov/IOTD/view.php?id=79677)

The flow hydrodynamics drives mixing processes in large rivers, affecting water quality, sediment dynamics, and bed morphological change. The advection and dispersion of sediment particles is particularly complicated in the vicinity of groynes and other hydraulic structures that are introduced into rivers for control purposes.

[1,2,3] http://www.earthobservatory.nasa.gov

Field observations, laboratory studies, and numerical simulation all offer insights into advection and mixing processes in large rivers. The resulting knowledge is useful to hydraulic engineers charged with designing works to control flood flows, protect riverbanks against erosion, and optimize flood protection works. Ongoing advances in computer power and mathematical modelling techniques, allow numerical simulation techniques to be used increasingly to model water and sediment flows and mixing processes in rivers. The following section presents a brief review of research undertaken to develop advanced numerical solvers of shallow flows in rivers.

1.3 Review of Literature

Over the past forty or so years, considerable effort has gone into the development of flow models that can accommodate complicated domains with curved and/or irregular boundaries. A variety of grid generation methods have been proposed including structured rectangular grids, structured curvilinear boundary-fitted grids, unstructured grids, sliding grids, and hierarchical grids; for example, a comprehensive introduction to block-structured and unstructured grids is given by Thompson *et al*. (1999).

1.3.1 Rectangular grid shallow flow solvers

The earliest numerical solvers of the shallow flows were based on structured rectangular grids (see e.g. Leendertse 1967; Abbott 1973; Kuipers and Vreugdenhil 1973; Hodgins 1977; Falconer 1980). Recent examples include the work by Stefanescu and Navon (2013) who present an alternating direction implicit (ADI) finite-difference solver for the shallow water equations (SWEs) on rectangular grids, that decreased the discretization error in time and space. Borsboom *et al.* (2014) applied an ADI SWEs solver to simulate the river Waal in Holland and hence design groynes by which to control the river when in flood. A major drawback of rectangular grids is that they do not conform to curved or irregular boundaries and this means that the grid can resemble a staircase at inclined boundaries – causing spurious reflections close to the boundary. As noted by Liang and Borthwick (2007), there are several ways to make a Cartesian grid conform to curved and irregular boundaries. A traditional approach is to use irregular stars in a finite difference scheme, such as

discussed by Sabersky *et al*. (1989) in the context of potential flow solvers. A more powerful approach is the Cartesian cut-cell method (see e.g. Yang *et al*. 1997; Causon *et al*. 2000, 2001, Liang *et al*. 2007), which involves representing the boundary as piece-wise continuous straight lines which is effectively cut out of the Cartesian template, and is discussed in Section 1.3.5. Another modern approach is the immersed boundary method (see e.g. Mohd-Yosuf, 1997; Ye *et al*., 1999; and Tseng and Ferziger, 2003). A simple treatment of curved boundaries in a finite volume shallow flow solver has been applied successfully by Liang and Borthwick (2008) to cases of a hydraulic jump in a channel with an inclined lateral wall, wind-induced flow in a circular domain, and shock diffraction in a contraction-expansion channel.

1.3.2 Curvilinear shallow flow solvers

Of the various methods available for grid generation, curvilinear structured grids are particularly appealing for shallow water applications in domains where the flow boundaries conform approximately to curves. Curvilinear grids are generated by mapping an arbitrary-shaped domain onto a simple rectangular shape, or a series of interconnected rectangles. For shallow flow simulation on such grids, the shallow water equations are rewritten in a transformed version using Jacobian mappings, and the resulting grids may be non-orthogonal or orthogonal depending on the transformation method used. The velocity components may be Cartesian or transformed into covariant or contravariant components. Thompson and his co-workers pioneered the use of curvilinear grid generation methods from the 1970s onwards. For example, Thompson *et al.* (1974) used finite differences to solve the discretized Poisson-type grid generation equations and produced grids for an arbitrary two-dimensional body. Thompson *et al*. then were able to solve the continuity and Navier-Stokes equations and simulate laminar flow past arbitrary airfoils. Following Thompson *et al*.'s research, Johnson and co-workers (Johnson 1980; Johnson 1982; and Johnson *et al*. 1982), Häuser *et al*. (1985, 1986) and Wijbenga (1985a, b) developed the first solvers of the shallow water equations on orthogonal grids. Willemse *et al.* (1985) introduced an orthogonal curvilinear shallow water equation solver whereby the insertion of additional geometric terms allowed greater mesh flexibility than for a non-orthogonal curvilinear solver. Borthwick and Barber (1992)

solved the non-orthogonal shallow water equations with Cartesian velocity components for jet-forced flow in a circular reservoir. In another study, Borthwick and Kaar (1993) solved a depth-averaged vorticity-transport form of the shallow water equations using curvilinear systems. Lin and Falconer (1995) solved shallow water and species transport equations on orthogonal curvilinear grids, using an alternating direction implicit scheme, to simulate sediment fluxes in estuarine waters. Borthwick and Akponasa (1997) solved the same problem using the non-orthogonal curvilinear shallow water equations rewritten in contravariant velocity components. Generally speaking, curvilinear shallow flow models are more desirable for simple flow domains with smoothly curving boundaries than those models with skewed mesh cells encountered in more complicated domains. Wu *et al.* (2004) used a non-orthogonal curvilinear shallow flow model to solve the depth averaged predict flow and pollutant transport in a bend of a meandering channel, with turbulence closure achieved using the depth-averaged k-ε equations. Tong and Meng (2006) solved the SWEs based on non-orthogonal curvilinear grids for flow in a 30^o flow bifurcation, and obtained predictions in excellent agreement with field data. George (2007) also solved the SWEs on non-orthogonal curvilinear grids to study tides in the Menai Strait, Wales, again achieving good agreement with observations. Jing *et al*. (2014) used a depth-averaged k-ε model with body-fitted coordinates to simulate flow in bends of the Yellow River, China, and reported good agreement between the predicted and observed water surface slopes.

1.3.3 Nested and patched grid shallow flow solvers

To predict shallow flow in complicated domains, researchers have also developed patched or nested rectangular grid models based on the shallow water equations. Hu *et al*. (2005) utilized mesh-patching to improve spatial resolution of a Godunov-type finite volume shallow flow solver with fluxes computed using an HLLC approximate Riemann solver. Hu *et al*. applied their model to simulate moving bores, circular dam break, oblique bore reflection, and the propagation of a bore wave over a hump. Hu *et al*. obtained adequate agreement between their results and alternative numerical simulations. St-Cyr *et al*. (2007) solved the SWEs on non-conforming adaptive and nested grids for simulating flow over a mountain and a steady-state

geostrophic flow. Hanrahan (2009) used nested and patched grids to simulate pollutant transport in a complex domain. Springer *et al.* (2009) simulated upwelling circulation and the transition zone off the Oregon coast on nested and patched grids; the model predicted the sea level over complex bed topography, jet separation and offshore eddy formation processes. A two-way nesting technique is incorporated in DIVAST, one of the leading shallow flow solvers (Falconer and Mardapitta-Hadjipandeli 1990; Falconer *et al*. 2001, Nash and Hartnett 2010, 2014).

1.3.4 Multi-grid shallow flow solvers

Multi-grid methods are a natural extension of nested and patched methods, and utilise a series of increasingly coarse meshes to solve elliptic equations very rapidly. Spitaleri and Corinaldesi (1997) used a multi-grid semi implicit finite difference algorithm to solve the two-dimensional shallow water equations and modelled surges, tides and Coriolis-induced free surface motions of a large-scale body of water in a basin. Spitaleri and Corinaldesi solved the water elevation gradient in the momentum equations and velocity divergence in the continuity equation by means of an Eulerian-Lagrangian method. The linear pentadiagonal discrete equations were solved using the multi-grid method. From their test results, Spitaleri and Corinaldesi observed that their method was highly flexible and applicable for physically large problems, and found that their multi-grid algorithm was faster than the conjugate gradient method. Later, Mitchell and Fulton (2000) formulated an adaptive multi-grid solver of the shallow water equations with mesh refinement, and used the solver to investigate cyclone tracking. Their results indicated that use of both multi-grid and mesh refinement can result in the same accuracy as on a uniform grid, but with substantially less computational work. Yost and Rao (2001) solved the shallow water equations in finite difference form on multi-grids with a MacCormack scheme to simulate open channel flow with strong shocks, including a hydraulic jump.

1.3.5 Cut-cell finite volume shallow flow models

The cut-cell finite volume method is a powerful alternative to other forms of boundary fitting in that it does not require curvilinear mappings; instead, solid regions are cut out of the underlying computational domain preventing a boundary staircase

from being created. The boundary therefore consists of a series of straight-line segments. Causon *et al*. (2000, 2001) pioneered use of the cut-cell computational method for solving shallow water problems with arbitrary complex geometries and moving boundaries. The model was validated for a ship's hull moving at supercritical velocity and two hypothetical landslides. Causon *et al*. (2001) found that cut-cells facilitated a boundary-fitted gridding capability suitable for solving shallow water flows in domains of particularly complicated geometry. By using a Riemann solver, Causon *et al*. were able to simulate accurately dam-breaks, hydraulic bore waves, and transitional flows involving sub/super-critical discontinuities. Liang *et al*. (2007) used a combination of cut-cells and quadtree grids to solve the shallow water equations, and demonstrated substantial computational efficiency gains in simulating surge wave reflection, inviscid flow past a cylinder at low Froude number, and bore-cylinder interaction. Bai *et al*. (2009) employed a finite volume solver and cut-cell method to simulate free surface wave propagation over a submerged dike and radiation waves induced by a circular cylinder. Kim *et al*. (2010) solved the shallow water equations on cut-cell grids to simulate a dam-break in a channel containing a 45^{o} bend, and obtained predictions in good agreement with experimental results. Using a similar method, Hyunuk and Soonyoung (2012) solved the shallow water equations on cut-cell grids to simulate flow in an inclined domain, free surface oscillations in a parabolic container, shock reflection in a circular domain, and urban flood inundation.

1.3.6 Godunov-type shallow flow models

Over the past 25 years, Godunov-type shock-capturing schemes have become increasingly popular for modelling steep-fronted, transcritical, even discontinuous flows. In such schemes, the shallow water equations are usually discretised using the finite volume technique and a Riemann solver used to model the movement of shock and rarefaction waves across cell interfaces. Comprehensive reviews are given by Toro (2001), LeVeque (2002), and Guinot (2003). Godunov-type shallow flow models have been developed on unstructured triangular meshes (see e.g. Anastasiou and Chan 1997) and on hierarchical quadtree grids (see e.g. Rogers *et al*. 2001, 2003, Liang *et al*. 2004, 2008, 2009). Shallow flow solvers based on adaptive quadtree grids have been applied to wind-induced flow in a basin and jet-forced flow in a reservoir (Rogers

et al. 2001), propagation of a wave over a hump (Rogers *et al*. 2003), dam- and dyke breaks (Liang *et al*. 2004), urban flooding (Liang *et al*. 2008), and wetting and drying (Liang and Borthwick, 2009). In implementing the finite volume method, the governing shallow water equations are written as a sum of a vector of temporal derivatives of dependent variables plus a vector of flux gradients minus a vector of source terms. The vector of flux gradients must balance that of source terms in order to ensure the finite volume scheme does not introduce spurious fluxes at what should be equilibrium conditions. Successful techniques for balancing the flux gradient and source terms include numerical balancing (see e.g. Vázquez-Cendón, 1999; LeVeque 1998; Hubbard and Garcia-Navarro 1999; Zhou *et al*. 2001) and mathematical algebraic balancing (see e.g. Bermudez and Vázquez, 1994; Rogers *et al*., 2001, 2003; and Liang and Borthwick, 2009). The advantage of Godunov-type finite volume solvers is that they provide accurate representations of complicated shallow flow phenomena, and they can be computationally very efficient (especially when solved on adaptive grids).

1.3.7 Lattice Boltzmann shallow flow solvers

Zhou (2004) introduced a unique solver known as LABSWE for shallow water equations based on the lattice Boltzmann method. This method includes Boltzmann equations with force terms or without force terms, using a lattice pattern, and utilizes equilibrium distribution functions. The lattice Boltzmann method for shallow water equations has been applied to the study of turbulent flow (LABSWE^{TM}). Zhou (2004) has successfully verified the LABSWE solver for many benchmark test cases including: steady flow over a bump, tidal flow over regular bed, flow around cylinder, steady flow over an irregular bed, tidal flow over an irregular bed, 2D steady flow over a bump, wind-induced circulation in a dish-shaped basin, flow in a strongly curved channel, and flow past multiple bodies. The LABSWE^{TM} solver was verified for turbulent flow in a rectangular channel and flow over a submerged island. Zhou and Borthwick (2011) applied the lattice Boltzmann shallow flow solver to simulate a flow fully mixed in the vertical direction but with variable density in the horizontal direction, and obtained excellent agreement with the analytical solution for hydrostatic equilibrium of liquid in a tank containing non-uniform bed terrain, and for horizontal

diffusion in a uniform flow field. Li *et al*. (2015) applied the lattice Boltzmann model to solve SWEs on complicated domain, and verified the model for flow over a bed hump and tidal (long wave) propagation over complicated bed topographies. Li *et al*. found that the discretization error in time and space for SWEs decreased by combining hydrostatic pressure and the bed slope.

1.3.8 Two and three-dimensional free surface flow solvers

Commercial software based on the complete two- and three-dimensional mass and momentum conservation equations have also been utilized for shallow flow modelling using finite difference, finite element, and finite volume techniques (see e.g. Stansby & Lloyd, 1995). Three-dimensional solvers can be categorised according to the way the grid is handled between the free surface and bed is handled: mappings using a sigma-transform, and fixed horizontal layers (for a more detailed explanation see e.g. Bourban 2013). Commercial software can also be categorised according to discretization technique. For example[1] FLOW-3D, ADCIRC, ICOM, FLUIDITY, HONEI, SWAMP, FESW and TELEMAC are based on finite element solvers of the shallow water continuity and momentum equations. Gerris, HEC-RAS2D, MIKE 3, ANUGA**,** SHALL 4, Delft3D and SULIS are finite-difference shallow flow solvers. CHAMMP, SWASHES, 4D-VAR, GEOCLAW, Full SWOF, BreZo, PYCLAW, PorAS, LIBMPDATA++ and OpenFOAM use finite volumes to discretize the shallow water equations (see e.g. Bourban 2013). The following examples indicate the wide range of applications of these codes. Vanderkimpen *et al*. (2009) used the MIKE suite of codes to develop a flood inundation model of the Belgian coast. Qiu and Fang (2009) applied FLOW-3D to simulate shallow flow in a channel containing a weir, a channel with complex bottom geometry, and a river reach containing a weir. They verified their results using laboratory measurements. Tung *et al*. (2015) used TELEMAC to simulate flow in the Mekong River, China.

There are also plenty of examples in the literature of custom-built 3-D solvers being applied to river flows. Lv and Feng (2011) developed a 3-D finite volume model simulation of flow around an *S*-shaped bend in the Yellow River, China, with

[1] http://www.swmath.org/?term=shallow%20water

turbulence closure provided by the *k-ε* equations. Kasvi *et al*. (2014) applied 3-D computational models to near-bed sediment transport in a meandering river bend and validated their results using field measurements.

A new generation of 3-D environmental free surface flow models is also being established using smoothed-particle hydrodynamics (SPH), a meshless technique that originated in astrophysics (Lucy 1977; Gingold and Monaghan 1977), and has since been extensively applied in computational fluid and solid mechanics (see e.g. Liu and Liu, 2003). Over recent years, the SPHysics series of open-source codes has been developed by researchers at Johns Hopkins University, the University of Vigo, the University of Manchester, and the University of Rome La Sapienza, and applied to a very wide variety of environmental free surface flow problems, shallow flow inundation (see e.g. Gómez-Gesteira *et al*. 2010; Vacondio *et al*. 2013). A recent example of the application of smoothed-particle hydrodynamics to fluvial processes is provided by Chen *et al*. (2015) who utilized smooth-particle hydrodynamics to simulate sediment transport in a *U*-shaped channel and in the Liusha River, China.

1.3.9 Lagrangian particle tracking in shallow flows

Lagrangian particle tracking is of great use in investigating dispersion processes in advection-dominated flow fields. The seminal work on chaotic advection is due to Aref (1984) who examined the very complicated trajectories of particles in the simple, but abruptly alternating flow field of a pair of blinking vortices. Aref discovered that the particle motions altered from periodic to chaotic as the stirring strength increased, and used stroboscopic maps (i.e. snapshots of particle positions at periodic intervals, where the period matches the blinking period of the vortices) to visualize the influence of stirring period on the tendency towards chaos.

Kranenburg (1992) considered the wind-induced chaotic advection of particles in a closed circular dish-like basin, where the wind direction changed periodically, and the flow field altered abruptly. Kranenburg's model used a theoretical stream function distribution that fitted the solution to the shallow water equations to estimate the depth-averaged velocity field driven by the wind at steady-state in a circular basin. Although the velocity field is not realistic – the depth-averaging obscures three-dimensional flow

processes in this case – the progress from periodic to chaotic advection is remarkably similar to that obtained by Aref for the blinking vortex problem, noting that in Kranenburg's case the wind strength replaces the stirrer strength as the governing parameter.

It is also worth noting that there are many examples of particle tracking used to determine contaminant dispersion in shallow flows. Pearson and Barber (1998) applied Lagrangian particle tracking to simulate the dispersion of coastal pollutant by solving an advection-diffusion equation on orthogonal boundary-fitted grids. Weitbrecht *et al* (2003) used 2-D particle tracking to examine 1D advection-diffusion equation in a river, and validated their model against laboratory measurements of flow in a rectangular open channel and flow in a rectangular channel flow with groynes.

In a study that is particularly relevant to the present thesis, Zsugyel *et al*. (2014) utilized particle tracking to investigate mixing processes in the vicinity of groynes in the Danube River, Hungary. Zsugyel *et al*. verified their model for particle trajectories in the vicinity of a single groyne in a rectangular channel, and validated their model against velocimetry measurements of particle motions in a laboratory basin. Zsugyel *et al*. examined the nonlinear characteristics of particle trajectories, and present information on flushing times, particle escape rates, mixing processes (through Lyapunov exponents), the presence of Lagrangian coherent structures, chaotic saddles, and fractal dimensions.

1.3.10 Summary of findings

The foregoing review indicates the large number of different ways in which river flow simulation may be undertaken numerically. Typical methods of solving the depth-averaged shallow water equations include: alternating-direction implicit (ADI) finite-difference solvers on rectangular, patched, and nested Cartesian grids; multi-grid solvers; curvilinear boundary-fitted techniques and cut–cell approaches to handling complicated shallow flow geometries; Godunov-type shallow flow models on adaptive hierarchical grids for application to steep-fronted trans-critical flows in complicated domains; lattice Boltzmann shallow flow solvers; and high-order discontinuous Galerkin finite element solvers on unstructured grids. In 3-D, mapped,

layered, volume-of-fluid, and level-set techniques have been developed to solve environmental free surface flows on Eulerian grids. There are also rapid developments in Lagrangian approaches, such as smoothed-particle hydrodynamics (SPH).

The present study aims at investigating mixing processes at the bend of a large river. Therefore, a curvilinear system approach will be taken, given the curved boundary geometry and curved flow lines. Particle tracking methods will be used to model mixing processes.

1.4 Aim and Objectives

The aim of the research contained in this thesis is to develop a curvilinear systems based model of shallow flows in wide rivers coupled to a Lagrangian particle tracking model that can be used to investigate advection and mixing processes in a heavily groyned river. The main objectives are

- To develop an efficient numerical technique for curvilinear grid generation; this will involve comparison of the computational performance of three iterative techniques (successive over-relaxation, multi-grid solver, and conjugate gradient) used to solve Poisson-type grid generation equations.
- To develop numerical solvers of both the Cartesian and curvilinear shallow water equations using second-order central differences in space and an explicit time marching scheme based on either second-order Adams-Bashforth or fourth-order Runge-Kutta schemes.
- To verify the shallow flow solvers using standard test cases including: uniform flow in a rectangular channel, wind-induced set up in rectangular and circular basins, recirculation in a channel with a side-wall expansion, and Shallow flow in a rectangular channel with single groyne.
- To develop a Lagrangian particle tracking solver of chaotic advection of particles, and verify the model for the cases of particle motions in the flow fields of a free vortex and a pair of blinking vortices.

- To simulate the shallow flow hydrodynamics and the advection of trace particles along a river bend containing groynes that is representative of an idealised reach of the River Danube, Hungary. The presence of the groynes is modelled using enhanced bed roughness.

1.5 Thesis Outline

The thesis is structured as follows. Chapter 1 provides an introduction to wide rivers and the use of groynes for flood protection followed by a review of literature concerning the modelling of shallow flows in wide rivers, and on mixing processes. Chapter 2 focuses on the development of iterative numerical grid generators based on Poission-type equations and presents results from verification cases. Chapter 3 presents a derivation of the governing shallow flow equations in both Cartesian and curvilinear forms, and describes their discretization and numerical solution using second-order finite differences in space and second-order Adams-Bashforth and fourth-order Runge-Kutta schemes in time. Chapter 4 describes benchmark test cases to verify Cartesian and curvilinear shallow water equations solver. Chapter 5 presents details of the Lagrangian particle tracking scheme used to represent advection and mixing processes, and its verification and validation using standard benchmark problems (such as particle trajectories driven by a free vortex and a pair of blinking vortices.) Chapter 6 presents a demonstration case of particle advection and mixing in the vicinity of a groyne in a river bend representative of a reach of the Danube River in Hungary. Chapter 7 concludes the thesis and suggests future directions that could be taken arising from the present research.

Chapter 2

Grid Generation

Chapter 2 describes the curvilinear boundary-fitted method for grid generation that will be used in later chapters to provide grids for the shallow flow solver. The method involves mapping arbitrary boundaries from the Cartesian system onto a rectangle in a transformed system using Poisson-type elliptic partial differential equations. A second-order central finite difference method is used to solve these Poisson-type grid generation equations by means of an iterative method. Three iteration schemes are considered: successive over-relaxation (SOR), multi-grid (MG), and conjugate gradient (CG). Control terms are included in the Poisson-type grid generation equations in order to produce distorted grids. The performance of these three iterative methods is assessed for various benchmark grids.

2.1 Governing Grid Generation Equations

A curvilinear boundary-fitted approach is used to map a physical domain from the Cartesian system onto a rectangle in a transformed system. Figure 2.1 shows an arbitrary original physical domain and its transformed rectangular domain. The boundary of the domain is split into four parts: named S_1, S_2, S_3, S_4 in the Cartesian system and T_1, T_2, T_3, T_4 in the transformed system. Along T_1, ξ takes various values, but η is constant. Along T_2, ξ takes different values and η is constant. Along T_3, η takes different values and ξ is constant; similar conditions hold at the T_4 boundary. Following Thompson *et al.* (1974), the following Poisson equations are used to transform the physical domain S in Cartesian coordinates onto a transformed rectangular domain T in general curvilinear coordinates:

$$\frac{\partial^2 \xi}{\partial x^2} + \frac{\partial^2 \xi}{\partial y^2} = P(x, y) \qquad (2.1)$$

and

$$\frac{\partial^2 \eta}{\partial x^2} + \frac{\partial^2 \eta}{\partial y^2} = Q(x, y) \qquad (2.2)$$

where x and y are Cartesian co-ordinates in the physical plane, ξ and η are curvilinear coordinates in the transformed plane, and P and Q are expressions for concentrating ξ – lines and η – lines, respectively. The Jacobian method is applied in order to transform the governing Poisson equations from the Cartesian system to a transformed system where the boundaries form a rectangle in transformed system. Letting $x = f(\xi,\eta)$ and $y = g(\xi,\eta)$, then the Jacobian of x and y with respect to ξ and η is denoted by $J = \partial(x,y)/\partial(\xi,\eta)$

$$J = \frac{\partial(x,y)}{\partial(\xi,\eta)} = \begin{vmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial x}{\partial \eta} \\ \frac{\partial y}{\partial \xi} & \frac{\partial y}{\partial \eta} \end{vmatrix} = \frac{\partial x}{\partial \xi}\frac{\partial y}{\partial \eta} - \frac{\partial x}{\partial \eta}\frac{\partial y}{\partial \xi} \qquad (2.3)$$

Note that, if $x = f(\xi,\eta)$ and $y = g(\xi,\eta)$, is a transformation of coordinates, then

$$\iint_R F(x,y)\,\mathrm{d}x\,\mathrm{d}y = \iint_S F\big(f(\xi,\eta), g(\xi,\eta)\big)\,J\,\mathrm{d}\xi\mathrm{d}\eta \qquad (2.4)$$

where J is the transformation Jacobian. Consider mapping from the physical plane onto a rectangle, as illustrated in Figure 2.1. If $\xi = f_1(x,y)$ and $\eta = g_1(x,y)$, then the Jacobian of ξ and η with respect to x and y denoted by $J_1 = \partial(\xi,\eta)/\partial(x,y)$ is:

$$J_1 = \frac{\partial(\xi,\eta)}{\partial(x,y)} = \begin{vmatrix} \frac{\partial \xi}{\partial x} & \frac{\partial \xi}{\partial y} \\ \frac{\partial \eta}{\partial x} & \frac{\partial \eta}{\partial y} \end{vmatrix} = \frac{\partial \xi}{\partial x}\frac{\partial \eta}{\partial y} - \frac{\partial \xi}{\partial y}\frac{\partial \eta}{\partial x} \qquad (2.5)$$

The transformations are reversible and so each Jacobian has to be the inverse of the other. That is, $J^{-1}=J_1$. Thus,

$$J^{-1} = \frac{1}{J}\begin{vmatrix} \frac{\partial y}{\partial \eta} & -\frac{\partial x}{\partial \eta} \\ -\frac{\partial y}{\partial \xi} & \frac{\partial x}{\partial \xi} \end{vmatrix} = \begin{vmatrix} \frac{\partial \xi}{\partial x} & \frac{\partial \xi}{\partial y} \\ \frac{\partial \eta}{\partial x} & \frac{\partial \eta}{\partial y} \end{vmatrix} \qquad (2.6)$$

By inspection:

$$\frac{\partial \xi}{\partial x} = \frac{1}{J}\frac{\partial y}{\partial \eta} \quad or \quad \xi_x = \frac{y_\eta}{J} \qquad \frac{\partial \xi}{\partial y} = -\frac{1}{J}\frac{\partial x}{\partial \eta} \quad or \quad \xi_y = -\frac{x_\eta}{J} \qquad (2.7a)$$

$$\frac{\partial \eta}{\partial x} = -\frac{1}{J}\frac{\partial y}{\partial \xi} \quad or \quad \eta_x = -\frac{y_\xi}{J} \qquad \frac{\partial \eta}{\partial y} = \frac{1}{J}\frac{\partial x}{\partial \xi} \quad or \quad \eta_y = \frac{x_\xi}{J} \qquad (2.7b)$$

Hence,

$$\xi_{xx} = \frac{\partial}{\partial x}(\xi_x) = \frac{\partial}{\partial x}\left(\frac{y_\eta}{J}\right) = \frac{\partial}{\partial \xi}\left(\frac{y_\eta}{J}\right)\frac{\partial \xi}{\partial x} + \frac{\partial}{\partial \eta}\left(\frac{y_\eta}{J}\right)\frac{\partial \eta}{\partial x} \ ,$$

$$\xi_{xx} = \frac{\xi_x y_{\xi\eta} + \eta_x y_{\eta\eta}}{J} - \frac{{\xi_x}^2 J_\xi + \xi_x \eta_x J_\eta}{J} \qquad (2.8)$$

$$\xi_{yy} = \frac{\partial}{\partial y}(\xi_y) = \frac{\partial}{\partial y}\left(-\frac{x_\eta}{J}\right) = -\frac{\partial}{\partial \xi}\left(\frac{x_\eta}{J}\right)\frac{\partial \xi}{\partial y} + \frac{\partial}{\partial \eta}\left(\frac{x_\eta}{J}\right)\frac{\partial \eta}{\partial y} \ ,$$

$$\xi_{yy} = \frac{-\xi_y x_{\xi\eta} - \eta_y x_{\eta\eta}}{J} - \frac{{\xi_y}^2 J_\xi - \xi_y \eta_y J_\eta}{J} \qquad (2.9)$$

$$\eta_{xx} = \frac{\partial}{\partial x}(\eta_x) = \frac{\partial}{\partial x}\left(-\frac{y_\xi}{J}\right) = -\left[\frac{\partial}{\partial \xi}\left(\frac{y_\xi}{J}\right)\frac{\partial \xi}{\partial x} + \frac{\partial}{\partial \eta}\left(\frac{y_\xi}{J}\right)\frac{\partial \eta}{\partial x}\right] \ ,$$

$$\eta_{xx} = -\frac{\xi_x y_{\xi\xi} + \eta_x y_{\xi\eta}}{J} - \frac{{\eta_x}^2 J_\eta + \xi_x \eta_x J_\xi}{J} \qquad (2.10)$$

$$\eta_{yy} = \frac{\partial}{\partial y}(\eta_y) = \frac{\partial}{\partial y}\left(\frac{x_\xi}{J}\right) = \left[\frac{\partial}{\partial \xi}\left(\frac{x_\xi}{J}\right)\frac{\partial \xi}{\partial y} + \frac{\partial}{\partial \eta}\left(\frac{x_\xi}{J}\right)\frac{\partial \eta}{\partial y}\right] \ ,$$

$$\eta_{yy} = \frac{\xi_y x_{\xi\xi} + \eta_y x_{\xi\eta}}{J} - \frac{{\eta_y}^2 J_\eta + \xi_y \eta_y J_\xi}{J} \qquad (2.11)$$

$$\xi_{xy} = \xi_{yx} = \frac{\partial}{\partial y}(\xi_x) = \frac{\partial}{\partial y}\left(\frac{y_\eta}{J}\right) = \frac{\partial}{\partial \xi}\left(\frac{y_\eta}{J}\right)\frac{\partial \xi}{\partial y} + \frac{\partial}{\partial \eta}\left(\frac{y_\eta}{J}\right)\frac{\partial \eta}{\partial y} \ ,$$

$$\xi_{xy} = \frac{\xi_y y_{\xi\eta} + \eta_y y_{\eta\eta}}{J} - \frac{\xi_x \xi_y J_\xi + \xi_x \eta_y J_\eta}{J} \qquad (2.12)$$

and

$$\eta_{xy} = \eta_{yx} = \frac{\partial}{\partial y}(\eta_x) = \frac{\partial}{\partial y}\left(-\frac{y_\xi}{J}\right) = -\left[\frac{\partial}{\partial \xi}\left(\frac{y_\xi}{J}\right)\frac{\partial \xi}{\partial y} + \frac{\partial}{\partial \eta}\left(\frac{y_\xi}{J}\right)\frac{\partial \eta}{\partial y}\right] \ ,$$

$$\eta_{xx} = -\frac{\xi_y y_{\xi\xi} + \eta_y y_{\xi\eta}}{J} - \frac{\eta_x \eta_y J_\eta + \xi_y \eta_x J_\xi}{J} \qquad (2.13)$$

Equations (2.1) and (2.2) may be written:

$$\begin{cases} \xi_{xx} + \xi_{yy} = P \\ \eta_{xx} + \eta_{yy} = Q \end{cases}$$

The above are Poisson equations in the physical plane. Substituting for $\xi_{xx}, \xi_{yy}, \eta_{xx}, \eta_{yy}, \xi_x, \xi_y, \eta_x, \eta_y$, and rearranging we obtain:

$$J^2(Px_\xi + Qx_\eta) + (x_\eta^2 + y_\eta^2)x_{\xi\xi} - 2(x_\xi x_\eta + y_\xi y_\eta)x_{\xi\eta} + (x_\xi^2 + y_\xi^2)x_{\eta\eta} = 0$$

and

$$J^2(Py_\xi + Qy_\eta) + (x_\eta^2 + y_\eta^2)y_{\xi\xi} - 2(x_\xi x_\eta + y_\xi y_\eta)y_{\xi\eta} + (x_\xi^2 + y_\xi^2)y_{\eta\eta} = 0$$

or

$$\alpha x_{\xi\xi} - 2\beta x_{\xi\eta} + \gamma x_{\eta\eta} = -J^2(Px_\xi + Qx_\eta) \qquad (2.14a)$$

and

$$\alpha y_{\xi\xi} - 2\beta y_{\xi\eta} + \gamma y_{\eta\eta} = -J^2(Py_\xi + Qy_\eta) \qquad (2.14b)$$

in which

$$\alpha = x_\eta^2 + y_\eta^2 \ , \ \beta = x_\xi x_\eta + y_\xi y_\eta \ , \quad \gamma = x_\xi^2 + y_\xi^2 \ .$$

The elliptic grid generation equations (2.14a-2.14b) are solved as a boundary value problem after discretisation using second-order central differences. To this end, let the subscripts *i* and *j* denote nodal positions in the transformed mesh (Figure 2.2), and taking the transformed mesh incremental lengths $\Delta\xi$ and $\Delta\eta$ to be unity, equations (2.15a) to (2.15b) are given as:

x co-ordinate

$$x_{i,j} = \frac{1}{2(\alpha_{i,j} + \gamma_{i,j})}\left[x_{i+1,j}\left(\alpha_{i,j} + \frac{J_{i,j}^2 P_{i,j}}{2}\right) + x_{i-1,j}\left(\alpha_{i,j} - \frac{J_{i,j}^2 P_{i,j}}{2}\right) \right.$$

$$\left. + x_{i,j+1}\left(\gamma_{i,j} + \frac{J_{i,j}^2 Q_{i,j}}{2}\right) + x_{i,j-1}\left(\gamma_{i,j} - \frac{J_{i,j}^2 Q_{i,j}}{2}\right) - \frac{\beta_{i,j}}{2}\left(x_{i+1,j+1} - x_{i+1,j-1} - x_{i-1,j+1} + x_{i-1,j-1}\right)\right] \qquad (2.15a)$$

y co-ordinate

$$y_{i,j} = \frac{1}{2(\alpha_{i,j}+\gamma_{i,j})}\left[y_{i+1,j}\left(\alpha_{i,j} + \frac{J_{i,j}^2 P_{i,j}}{2}\right) + y_{i-1,j}\left(\alpha_{i,j} - \frac{J_{i,j}^2 P_{i,j}}{2}\right)\right.$$

$$\left. + y_{i,j+1}\left(\gamma_{i,j} + \frac{J_{i,j}^2 Q_{i,j}}{2}\right) + y_{i,j-1}\left(\gamma_{i,j} - \frac{J_{i,j}^2 Q_{i,j}}{2}\right) - \frac{\beta_{i,j}}{2}\left(y_{i+1,j+1} - y_{i+1,j-1} - y_{i-1,j+1} + y_{i-1,j-1}\right)\right] \quad (2.15b)$$

where

$$\alpha_{i,j} = \tfrac{1}{4}\left[(x_{i,j+1} - x_{i,j-1})^2 + (y_{i,j+1} - y_{i,j-1})^2\right],\ \beta_{i,j} = \tfrac{1}{4}\left[(x_{i,j+1} - x_{i,j-1})(x_{i,j+1} - x_{i,j-1}) + (y_{i,j+1} - y_{i-1,j})(y_{i,j+1} - y_{i,j-1})\right],$$

$$\gamma_{i,j} = \tfrac{1}{4}\left[(x_{i+1,j} - x_{i-1,j})^2 + (y_{i+1,j} - y_{i-1,j})^2\right],\ J_{i,j} = \tfrac{1}{4}\left[(x_{i+1,j} - x_{i-1,j})(y_{i,j+1} - y_{i,j-1}) - (x_{i,j+1} - x_{i,j-1})(y_{i+1,j} - y_{i-1,j})\right].$$

In order to generate the grid, Equations (2.15) have to be solved iteratively on a grid, such as indicated in Figure 2.2 where $\xi = i\Delta\xi$ with $i = 0, 1, 2, ..., i_{\max}$, and $\eta = j\Delta\eta$ with $j = 0, 1, 2, ..., j_{\max}$. It is usual to choose $\Delta\xi = \Delta\eta = 1$. Boundary values are specified as Dirichlet conditions for $(x_{i,0}, y_{i,0})$ and $(x_{i,j\max}, y_{i,j\max})$ for $i = 0, 1, 2, ..., i_{\max}$ and $(x_{0,j}, y_{0,j})$ and $(x_{i\max,j}, y_{i\max,j})$ for $j = 0, 1, 2, ..., j_{\max}$. An iterative scheme is then applied to solve Equations (2.15) starting from initial guessed values at the interior of the grid until a suitable tolerance is reached. The resulting (x, y) values within the interior domain define the physical grid corresponding to the (ξ, η) values in the computational domain.

2.2 Numerical Solution of Grid Generation Equations using Successive Over-Relaxation

In successive over-relaxation, Equations (2.15) are solved point by point throughout the domain to give.

$$x_{new_{i,j}} = \frac{1}{2(\alpha_{i,j}+\gamma_{i,j})}\left[x_{i+1,j}\left(\alpha_{i,j} + \frac{J_{i,j}^2 P_{i,j}}{2}\right) + x_{i-1,j}\left(\alpha_{i,j} - \frac{J_{i,j}^2 P_{i,j}}{2}\right)\right.$$

$$\left. + x_{i,j+1}\left(\gamma_{i,j} + \frac{J_{i,j}^2 Q_{i,j}}{2}\right) + x_{i,j-1}\left(\gamma_{i,j} - \frac{J_{i,j}^2 Q_{i,j}}{2}\right) - \frac{\beta_{i,j}}{2}\left(x_{i+1,j+1} - x_{i+1,j-1} - x_{i-1,j+1} + x_{i-1,j-1}\right)\right] \quad (2.16a)$$

and

$$y_{newi,j} = \frac{1}{2(\alpha_{i,j} + \gamma_{i,j})}\left[y_{i+1,j}\left(\alpha_{i,j} + \frac{J_{i,j}^2 P_{i,j}}{2} \right) + y_{i-1,j}\left(\alpha_{i,j} - \frac{J_{i,j}^2 P_{i,j}}{2} \right) \right.$$

$$\left. + y_{i,j+1}\left(\gamma_{i,j} + \frac{J_{i,j}^2 Q_{i,j}}{2} \right) + y_{i,j-1}\left(\gamma_{i,j} - \frac{J_{i,j}^2 Q_{i,j}}{2} \right) - \frac{\beta_{i,j}}{2}\left(y_{i+1,j+1} - y_{i+1,j-1} - y_{i-1,j+1} + y_{i-1,j-1} \right) \right] \quad (2.16b)$$

The updated values are then determined using

$$x_{new_{i,j}} = x_{i,j} + \omega(x_{new_{i,j}} - x_{i,j}) \quad \text{and} \quad y_{new_{i,j}} = y_{i,j} + \omega(y_{new_{i,j}} - y_{i,j}) \qquad (2.17)$$

where ω is an acceleration parameter, usually set between 1 and 1.8 (Roache, 1970). By tuning ω, it is possible to obtain enhanced computational performance using SOR over Gauss-Seidel iteration (see e.g. Kreysig, 2011), which in turn is much faster than Jacobi iteration (see e.g. Kreysig, 2011) for a given convergence criterion and a given grid. In the following example (Figure 2.3), solutions for *x* and *y* are estimated by applying the successive over-relaxation solver to create grids for a circular domain of radius 150 m. Grid convergence and computational performance are tested for grids of 16 x 16, 32 x 32, and 64 x 64 nodes. A FORTRAN program has been written from scratch by the author to generate the grids. The boundary values are simply the *x* and *y* coordinates of the perimeter of the circle, which is divided into equal segments. The initial guessed values of *x* and *y* for the internal nodes are determined using linear interpolation. The program is then run until a suitable convergence criterion is met. The final grids are displayed using Matlab software. Figure 2.3 shows the progressively finer 16 x 16, 32 x 32 and 64 x 64 grids obtained. Here, ω = 1.6 and tolerances is 1.0e-8. The final SOR CPU times for the computations on the 16 x 16, 32 x 32 and 64 x 64 circular grids are 0.015, 0.328 and 4.546 s.

2.3 Numerical Solution of Grid Generation Equations using the Multigrid Method

The multi-grid method is an efficient iterative method for solving boundary-value elliptic problems. The method uses smoothing and coarse-grid correction to improve computational performance, with the solution speeded up by solving the residual on

fine grids. The main reason for using the multi-grid method (MG) is that it decreases the lower error wave frequencies on coarser grids and smooths the high frequency error wave on the fine grids. This process is known as relaxation. In other words, by improving the residual on coarse and fine grids, the high and low-frequency errors decrease. The MG method was first introduced by Fedoranko in 1964 aimed at the high order solution of linear equation systems. Bachvalov (1966) applied multi-grids to solve the Poisson equation within a rectangle, using a network approximation to discretise the elliptic equation. Further advances were made by Astrakhantsev (1971), Nicolaides (1977), and others – and an excellent textbook explaining the method has been authored by Briggs (1987). Nowadays, the multi-grid method is routinely applied in numerical investigations in computational hydraulics, structural engineering, aerospace engineering, particle physics, and fluid mechanics. Li and Fleming (1997) developed a three dimensional multi-grid model for fully nonlinear water waves propagation. Yadlin and Caughey (1989) used the multi-grid method to solve the three dimensional Euler equation for inviscid, compressible flow past a swept wing. Tiriki (2014) studied shallow water equations for open channels with shock wave (hydraulic jump) using multi-grid (MG) method, and introduced a hybrid solution of MG and finite element method to solve SWEs iteratively. Tiriki found satisfactory agreement between the numerical predictions and the experimental data, confirming the ability of MG to simulate complex hydraulic phenomena such as dam break, hydraulic jump, and shock wave propagation.

Herein, the multi-grid method is applied to solve the discretised Poisson equations (2.15). To recap, in order to generate the curvilinear mesh efficiently, the solver must be accurate, rapid to implement, and convergent (with decreasing grid size). As the number of grid cells (or nodes) increases, simple iterative methods such as SOR become increasingly inefficient and may even lose accuracy and become non-convergent, unlike multi-grid iteration. In particular, the error associated with the largest wave length across the grid is reduced considerably on the coarser grids utilised by the multi-grid method. A brief explanation is first given of the basics of MG theory, following Briggs (1987). Consider the one dimensional equation and Figure 2.4:

$$\frac{\partial^2 u}{\partial x^2} + u = f \qquad (2.18)$$

where $u = 1$ at $x = 0$ and $\frac{\partial u}{\partial x} = 0$ at $x = L$. Using second-order finite-differences we obtain:

$$\frac{v_{i-1} - 2v_i + v_{i+1}}{\Delta x^2} + v_i = f_i \qquad (2.19)$$

with $v_0 = 0$ and $v_n = v_{n-1}$, and where v_i denotes an approximate solution and u_j is the exact solution. By allocating the boundary conditions to equation (2.19) we obtain:

for $i = 1$ $\quad \frac{v_0 - 2v_1 + v_2}{\Delta x^2} + v_1 = f_1 \quad$ or $\quad v_0 + (\Delta x^2 - 2)v_1 + v_2 = \Delta x^2 f_1$

$$\text{or} \quad (\Delta x^2 - 2)v_1 + v_2 = \Delta x^2 f_1$$

for $i = 2$ $\quad \frac{v_1 - 2v_2 + v_3}{\Delta x^2} + v_2 = f_2 \quad$ or $\quad v_1 + (\Delta x^2 - 2)v_2 + v_3 = \Delta x^2 f_2$

$$\vdots$$

for $i = n\text{-}1$ $\quad \frac{v_{n-2} - 2v_{n-1} + v_n}{\Delta x^2} + v_{n-1} = f_{n-1}$ or $v_{n-2} + (\Delta x^2 - 2)v_{n-1} + v_n = \Delta x^2 f_{n-1}$

But $v_n = v_{n-1}$, and so $v_{n-2} + (\Delta x^2 - 1)v_{n-1} = \Delta x^2 f_{n-1}$

Rewriting equation (2.19) in matrix form yields:

$$\begin{bmatrix} \Delta x^2 - 2 & 1 & 0 & \dots & 0 & 0 & 0 \\ 1 & \Delta x^2 - 2 & 1 & \dots & 1 & 0 & 0 \\ 0 & 1 & \Delta x^2 - 2 & \dots & 1 & 1 & 0 \\ & & & \ddots & & & \\ & & & & \ddots & & \\ & & & & & \ddots & \\ & & & & & & \ddots \\ 0 & 0 & 0 & \dots & 1 & 1 & \Delta x^2 - 2 \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \\ \vdots \\ \vdots \\ \vdots \\ v_{n-2} \\ v_{n-1} \end{bmatrix} = \Delta x^2 \begin{bmatrix} f_1 \\ f_2 \\ \vdots \\ \vdots \\ \vdots \\ f_{n-2} \\ f_{n-1} \end{bmatrix}$$

or

$$\boldsymbol{Kv} = \boldsymbol{f} \qquad (2.20)$$

To estimate $\boldsymbol{v}$ from equation (2.20), the error term is first defined as:

$$\boldsymbol{e} = \boldsymbol{u} - \boldsymbol{v} \qquad (2.21)$$

Since we cannot estimate the exact solution, the residual is used

$$\boldsymbol{r} = \boldsymbol{f} - \boldsymbol{K}\boldsymbol{v} \qquad (2.22)$$

where $\boldsymbol{K}$ is the matrix in equation (2.20). Gauss-Seidel relaxation is used to estimate the residual and $\boldsymbol{v}$.

The next coarse grid residual based on relaxation method is:

$$r_n^{2h} = r_{2n-1}^h \qquad \text{for } 1 < n < \frac{\Delta x}{2} + 1 \qquad (2.23)$$

where, n is the number of grid points and h refers to the grid size.

Next, the error on the finer grid is used to obtain an improved approximation for $\boldsymbol{v}$:

$$v_{i+1}^h = v_i^h + e^h \qquad (2.24)$$

Similarly, by transferring information from the coarse grid to fine grid, the residual is improved:

$$r^{2h} = kv^{2h} - f^{2h} \qquad (2.25)$$

Using the residual obtained from equation (2.25), $\boldsymbol{v}$ is approximated for the next fine grid:

$$Kv^{4h} = f^{4h} \qquad (2.26)$$

The same process is repeated, this time taking the initial guess for $\boldsymbol{v}^{4h}$ as zero. The analysis continues until the coarsest grid is reached. For instance, if the coarsest grid is split up into units of size $64h$, the calculations are continued until $\boldsymbol{v}^{64h}$ is estimated. Addition of this value to the version of $\boldsymbol{v}^{32h}$, gives an accurate approximation to $\boldsymbol{v}^{32h}$. But before this stage, the value of $\boldsymbol{v}^{64h}$ should be transferred to the next finest grid. The same analysis is carried out for $\boldsymbol{v}^{32h}$, i.e., by transferring $\boldsymbol{v}^{32h}$ to the finest grid and adding to $\boldsymbol{v}^{16h}$, so that a more accurate value for $\boldsymbol{v}^{16h}$ is calculated. This analysis is continued up to the finest grid. This form of multi-grid method is known as the V-cycle (see e.g. Briggs, 1987 and Turnbull, 1995).

Figures 2.5 show the grids obtained by solving the grid generation equations (2.14) using V-cycle multi-grid iteration mapping a circular physical domain of 300 m

diameter onto a rectangular computational domain. In this study tolerances is 1.0e-8. The resulting grids are in almost exact agreement with those produced previously using successive over-relaxation (Figure 2.3). The main advantage of multi-grids is that it permits much finer grids to be generated accurately and rapidly than is possible using successive over-relaxation. The multi-grid method requires less CPU time than SOR, with the final CPU times for 16 x 16, 32 x 32 and 64 x 64 multi-grid circular grids are 0.125, 0.328 and 1.031 s, respectively.

2.4 Numerical Solution of Grid Generation Equations using the Conjugate Gradient Method

This section examines use of the conjugate gradient (CG) method, another powerful iterative technique for solving elliptic equations such as the Poisson and Laplace equations. The conjugate gradient (CG) method was first proposed by Hestenes and Stiefel (1952) as an iterative method to solve linear systems (see e.g. O'Leary, 2009). Hestenes and Stiefel reported that the conjugate gradient method had the following properties: the method is relatively simple to implement and is effective regarding CPU storage requirements; it has a high rate of convergence; the algorithm is stable with the error monotonically decreasing iteration by iteration, until the approximate solution is achieved. Young and Kang (1980) introduced the idealized generalized conjugate-gradient (IGCG) technique which increased the rate of convergence and improved the CPU time. Barragy *et al*. (1993) applied the conjugate gradient method to solve the nonlinear shallow water equations iteratively on curvilinear grids generated using the Helmholtz equation, and thus simulate tidal flows involving Coriolis acceleration. Barragy *et al*. demonstrated the advantage of CG method over other iterative methods such as Jacobi iteration. Paglieri *et al*. (1997) investigated parallelised computational solutions of the shallow water equations using the finite element scheme with conjugate gradient iteration and Schwarz preconditioning. Paglieri *et al*. found that the parallelised CG method was highly efficient computationally. Bonaventura and Rosatti (2002) solved the two-dimensional shallow water equations for coastal hydrodynamics on a structured locally refined Cartesian

grid using semi-implicit, mass conservative discretization scheme with a conjugate gradient algorithm.

Following Hestenes and Stiefel (1952), the CG method can be illustrated by considering the following system of linear equations:

$$K_{11}v_1 + K_{12}v_2 + \cdots + K_{1n}v_n = f_1$$

$$K_{21}v_1 + K_{22}v_2 + \cdots + K_{2n}v_n = f_2 \qquad (2.27)$$

$$\ldots \quad \ldots \quad \ldots \quad \ldots \quad \ldots \quad \ldots$$

$$K_{n1}v_1 + K_{n2}v_2 + \cdots + K_{nn}v_n = f_n$$

The above equations are written in matrix format:

$$\begin{bmatrix} k_{11} & k_{12} & \ldots & k_{1n} \\ k_{21} & k_{22} & \ldots & k_{2n} \\ k_{31} & k_{32} & \ldots & k_{3n} \\ & & \ddots & \\ & & \ddots & \\ k_{n1} & k_{n2} & \ldots & k_{nn} \end{bmatrix} \begin{bmatrix} V_1 \\ V_2 \\ \vdots \\ \vdots \\ V_n \end{bmatrix} = \begin{bmatrix} f_1 \\ f_2 \\ \vdots \\ \vdots \\ f_n \end{bmatrix}$$

or

$$\boldsymbol{Kv} = \boldsymbol{f} \qquad (2.28)$$

where $\boldsymbol{K}$ is a known, positive-definite matrix, and $\boldsymbol{v}$ and $\boldsymbol{f}$ are unknown and known vectors. The solution of $\boldsymbol{Kv} = \boldsymbol{f}$ is determined as $\boldsymbol{u}$, which satisfies $\boldsymbol{Ku} = \boldsymbol{f}$. If $\boldsymbol{v}$ is an estimation solution of $\boldsymbol{u}$, the difference $\boldsymbol{f} - \boldsymbol{Kv}$ is known as the residual vector $\boldsymbol{r}$.

$$\boldsymbol{r} = \boldsymbol{f} - \boldsymbol{Kv}$$

And the error vector for CG is obtained by:

$$\boldsymbol{e} = \boldsymbol{u} - \boldsymbol{v}$$

In implementation, the conjugate gradient CG method involves the following steps:
Step 1: Choose an arbitrary value for v_0 as the first estimate solution of u_0 and compute values for r_0 and q_0 from

$$q_0 = r_0 = f - kv_0 \qquad (2.29)$$

Step 2: Compute.

$$\alpha_i = \frac{|r_i|^2}{(q_i k q_i)} \qquad \text{or} \qquad \alpha_i = \frac{(q_i r_i)}{(q_i k q_i)} \tag{2.30}$$

$$v_{i+1} = v_i + \alpha_i q_i \tag{2.31}$$

$$r_{i+1} = r_i - \alpha_i k q_i \tag{2.32}$$

$$\beta_i = \frac{|r_{i+1}|^2}{|r_i|^2} \quad \text{or} \quad \beta_i = -\frac{(r_{i+1} k q_i)}{(q_i k q_i)} \tag{2.33}$$

Step 3: Approximate v_i as an estimate of the solution u_i, residual r_i, q_i and computation of v_{i+1} and r_{i+1} for the next iteration. In this stage v_i is closer to u_i after each iteration. This third step is commonly known as steepest descent, and is at the heart of the CG technique. Steps 2 and 3 are repeated until the grid generation process based on CG is completed. Figure 2.6 show the circular physical and mapped rectangular grids obtained by solving Equations (2.14) with the conjugate gradient method for a circular domain of diameter 300 m. Here tolerances is 1.0e-12 The resulting grids are almost identical to those obtained using the successive over-relaxation and multi-grid methods considered previously (see Figures 2.3 and 2.5, respectively). This confirms the conjugate gradient technique has been implemented correctly. In this study, the final CPU times required to generate 16 x 16, 32 x 32 and 64 x 64 circular grids using CG are 0.093, 0.265, and 1.011 s, respectively.

2.5 Grid generation control

By introducing suitable terms into the governing Poisson-type grid generation equations it is possible to control the local grid line intensity. This control is important so that we can set the spacing of lines and grids to be finer in areas of interest, such as near boundaries or where the gradients of physical variables are largest. We know that Equations (2.1) and (2.2) without *P* and *Q* reduce to Laplace's equation. The resulting transformed grid is relatively uniform and undistorted. However, the grids generated can be distorted in a controlled fashion by setting values to *P* and *Q* in the Poisson equation. This section describes this approach, following the methodology first introduced by Thompson *et al*. (1977). Based on Thompson *et al*.'s (1977) study, the

Laplace and Poisson elliptic partial differential equations are used to illustrate grid generation control.

$$\begin{cases} \xi_{xx} + \xi_{yy} = 0 \\ \eta_{xx} + \eta_{yy} = 0 \end{cases} \qquad (2.34)$$

$$\begin{cases} \xi_{xx} + \xi_{yy} = P(\xi, \eta) \\ \eta_{xx} + \eta_{yy} = Q(\xi, \eta) \end{cases} \qquad (2.35)$$

Assume $\acute{\xi}$ and $\acute{\eta}$ are solutions of Laplace equation (2.34), and $\acute{\acute{\xi}}$ and $\acute{\acute{\eta}}$ are the solutions of Poisson equation (2.35). Since Laplace and Poisson equations are elliptic equations, $\xi(x, y)$ and $\eta(x, y)$ will become harmonic on the boundary, which means $\acute{\acute{\xi}}$ and $\acute{\acute{\eta}}$ are sub-harmonic solutions on the boundary. This leads to an important conclusion, that is, for any constant *R*, the $\acute{\acute{\xi}} = R$ line approaches closer to $\acute{\acute{\xi}} = \xi_{max}$ than the $\acute{\xi} = R$ line. Meanwhile, if $\acute{\xi} = \xi_{max}$ is considered as the section cut at a boundary, then this can translate on the ξ axes, until $\acute{\acute{\xi}} = \xi_{max}$. By applying this procedure for η coordinate lines, grid generation control is established. (see e.g. Thompson *et al.* 1974).

2.5.1 Distorted grid based on attraction or repulsion to one side of a single line

In order to attract or repulse grid lines to or from one side of a single line in the physical domain corresponding to either $\xi = \xi_i$ or $\eta = \eta_j$, depending on the coordinate direction of interest, we set,

$$P = -a \exp\left(-d(\xi - \xi_i)\right) \qquad (2.36a)$$

and

$$Q = -a \exp\left(-d(\eta - \eta_j)\right) \qquad (2.36b)$$

where *a* is an amplitude and *d* is a decay factor. Equation (2.36*a*) causes ξ −constant lines to be attracted to the $\xi = \xi_i$ line, Similarly, Equation (2.36*b*) causes η −constant lines to be attracted to the $\eta = \eta_j$ line. Here, *a* and *d* control the grid system; the larger the values of *a* and *d*, the more the displacement of the lines. However, the values of

these two coefficients are limited; otherwise the grid lines in the physical domain may be either too distorted or insufficiently distorted.

2.5.2 Distorted grid based on attraction or repulsion to both sides of a single line

In order to attract or repulse grid lines to both sides of a prescribed line at either $\xi = \xi_i$ or $\eta = \eta_j$, the grid attraction operators are specified as:

$$P = -a\ \mathrm{sgn}(\xi - \xi_i)\exp\left(-d(\xi - \xi_i)\right) \qquad (2.37a)$$

and

$$Q = -a\ \mathrm{sgn}\left(\eta - \eta_j\right)\exp\left(-d\left(\eta - \eta_j\right)\right) \qquad (2.37b)$$

where sgn(x) is 1 , -1 or 0 depending on whether x is positive, negative or zero. Sgn creates discontinuities in the P and Q equations.

2.5.3 Multiple line attraction

In order to implement multiple line attraction (or repulsion) about $\xi = \xi_i$ for $i = 1 \ldots i_{\max}$ and $\eta = \eta_j$ for $j = 1 \ldots j_{\max}$, we use

$$P = -\sum_i a\ \mathrm{sgn}(\xi - \xi_i)\exp\left(-d(\xi - \xi_i)\right) \qquad (2.38a)$$

and

$$Q = -\sum_j a\ \mathrm{sgn}\left(\eta - \eta_j\right)\exp\left(-d\left(\eta - \eta_j\right)\right) \qquad (2.38b)$$

Herein, based on multiple line attraction, three lines are selected located towards the left, middle and right of the domain.

2.5.4 Multiple point attraction

The equations used to control grid generation based on multiple point attraction are:

$$P = -\sum_i \sum_j a \, \mathrm{sgn}(\xi - \xi_i) \exp(-d \sqrt{(\xi - \xi_i)^2 + (\eta - \eta_j)^2}) \qquad (2.39a)$$

and

$$Q = -\sum_i \sum_j a \, \mathrm{sgn}(\eta - \eta_j) \exp(-d \sqrt{(\xi - \xi_i)^2 + (\eta - \eta_j)^2}) \qquad (2.39b)$$

Equations (2.39) cause ξ −constant lines and η −constant lines to become attracted to specific points (ξ_i, η_j).

2.5.5 Demonstration cases for distorted grids

Figures 2.7 and 2.8 present examples of distorted grids obtained for the different control approaches within a rectangular domain obtained using SOR and MG on 16 x 16 and 32 x 32 grids. In all cases, the amplitude is a = 0.0015, decay factor d = 0.1, and rectangular physical domain dimensions are 300 m long by 150 m wide. The tolerance used to ascertain convergence is 1 x 10^{-8} in both the SOR and MG algorithms. Table 2.1 lists values for the lines ξ_i and η_j used for grid distortion within circular and rectangular domains based on the attraction or repulsion to one side of a single line, and to both sides of a single line. Table 2.2 provides details of the choice of lines for multiple line attraction considered here. Table 2.3 lists the values of (ξ_i, η_j) selected for the multiple point attraction tests. From Figures 2.7 and 2.8 it can be seen that there is generally good agreement between the results obtained using the SOR and MG algorithms. The results for single line attraction on one side and both sides, multiple line attraction and multiple point attraction indicate that SOR and MG provide converged grids that are very similar. Also, Figures 2.9 and 2.10 shows the comparison between CG and SOR methods.

The standard deviation of the difference between the grid coordinate values is used to evaluate the agreement between the SOR, MG and CG results for a case where a = 0.00045 and d = 0.1. The standard deviation formula used to compare SOR and MG is:

$$\sigma = \sqrt{\frac{1}{N} \sum [(\xi_{SOR} - \xi_{MG})^2 + (\eta_{SOR} - \eta_{MG})^2]} \qquad (2.40)$$

where σ is standard deviation, N is number of grids, ξ_{SOR} , η_{SOR} and ξ_{MG} , η_{MG} are grid values in the grid control studies, which are determined using the SOR and MG programs. Also, The same process is applied to approximate the standard deviation (σ) for CG and MG.

Table 2.4 lists the standard deviations between grid coordinates obtained between the SOR and MG schemes for various distortion approaches on a 16 x 16 grid; Table 2.4 also lists the corresponding standard deviations between the CG and MG schemes. Table 2.5 lists the results obtained for the finer 32 x 32 grid. The results indicate that grid generation methods provide similar estimates of the grid coordinates, with the CG and MG estimates generally closer to each other than to the SOR estimates. Table 2.6 compares the CPU time used by the SOR, MG, and CG algorithms to produce various distorted meshes in a 32 x 32 rectangular domain. It can be seen that the CG is the fastest, with MG of similar order to SOR for the cases selected.

2.6 Conclusion

The chapter has outlined the curvilinear grid generation method that will be used later in the thesis. The generation technique involves solving numerically a pair of elliptic partial differential equations to map a physical domain onto a rectangular computational domain. Three iterative methods have been considered; namely, successive over-relaxation (SOR), multi-grid (MG) and conjugate gradient (CG). It has been found that all three techniques produce perfectly adequate grids for a test circular domain, and that they can also produce distorted grids according to specified functions that are inserted into the grid generation equations. It is noted that SOR ceases to be a satisfactory technique for very fine grids because of the cumulative error across the grid, whereas both CG and MG do not possess this drawback. It is also confirmed that CG and MG work substantially faster than SOR, as would be expected.

Table 2.1 Grid control: single lines of attraction

rectangular grid (16 X 16)		rectangular grid (32 X 32)		rectangular grid (64 X 64)	
$\xi_i = 8$	$\eta_j = 8$	$\xi_i = 16$	$\eta_j = 16$	$\xi_i = 32$	$\eta_j = 32$

Table 2.2 Grid control: multiple lines of attraction

rectangular grid (16X16)		rectangular grid (32X32)		rectangular grid (64X64)	
$\xi_{i1} = 1$ $\xi_{i2} = 8$ $\xi_{i3} = 16$	$\eta_{j1} = 1$ $\eta_{j2} = 8$ $\eta_{j3} = 16$	$\xi_{i1} = 1$ $\xi_{i2} = 16$ $\xi_{i3} = 32$	$\eta_{j1} = 1$ $\eta_{j2} = 16$ $\eta_{j3} = 32$	$\xi_{i1} = 1$ $\xi_{i2} = 32$ $\xi_{i3} = 64$	$\eta_{j1} = 1$ $\eta_{j2} = 32$ $\eta_{j3} = 64$

Table 2.3 Grid control: multiple attraction points

rectangular grid (16X16)	rectangular grid (32X32)	c rectangular grid (64X64)
$(\xi_{i1}, \eta_{j1}) = (1,1)$ $(\xi_{i2}, \eta_{j2}) = (8,8)$ $(\xi_{i3}, \eta_{j3}) = (16,16)$	$(\xi_{i1}, \eta_{j1}) = (1,1)$ $(\xi_{i2}, \eta_{j2}) = (16,16)$ $(\xi_{i3}, \eta_{j3}) = (32,32)$	$(\xi_{i1}, \eta_{j1}) = (1,1)$ $(\xi_{i2}, \eta_{j2}) = (32,32)$ $(\xi_{i3}, \eta_{j3}) = (64,64)$

Table 2.4 Standard deviation of results obtained for distorted 16 x 16 grids generated within a rectangular domain (a = 0.00045, d = 0.1)

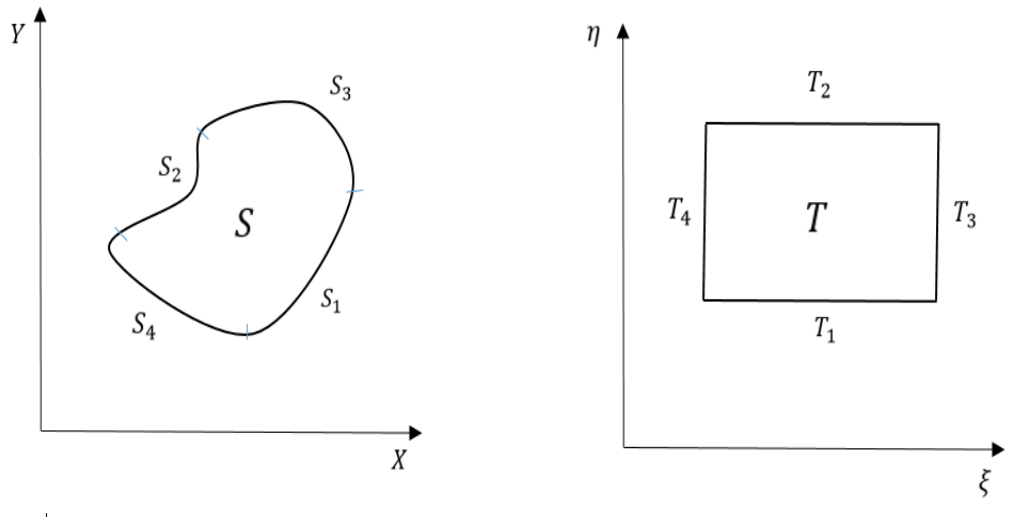
SOR & MG Single line attraction on one side	0.228629136226	CG & MG Single line attraction on one side	0.829741161673
SOR & MG Single line attraction on both sides	2.435856793151E-02	CG & MG Single line attraction on both sides	0.467631029867
SOR & MG multiple line attraction on both sides	2.109778469681E-02	CG & MG multiple line attraction on both sides	0.169367865260
SOR & MG multiple point attraction on both sides	2.212227880938E-02	CG & MG multiple point attraction on both sides	0.231639016354

Table 2.5 Standard deviation of results obtained for distorted 32 x 32 grids generated within a rectangular domain (a = 0.00045, d = 0.1)

SOR & MG Single line attraction on one side	1.63658793737	CG & MG Single line attraction on one side	1.67125374566
SOR & MG Single line attraction on both sides	0.401343781954	CG & MG Single line attraction on both sides	0.450638145016
SOR & MG multiple line attraction on both sides	9.162251626413E-02	CG & MG multiple line attraction on both sides	9.945381982959E-02
SOR & MG multiple point attraction on both sides	0.209015297980	CG & MG multiple point attraction on both sides	0.222022031412

Table 2.6 CPU time used to produce a 32 x 32 distorted rectangular grid using SOR, MG and CG schemes

Distorted Mesh Case	SOR	MG	CG
Single line attraction on one side	0.3120020	0.20280130	4.680030E-2
Single line attraction on both sides	0.29640190	0.21840140	6.240040E-2
Multiple line attraction on both sides	0.4368028	0.29640190	0.17160110
Multiple point attraction on both sides	0.42120270	0.3120020	7.80005E-2



(a) (b)

Figure 2.1 Curvilinear grid generation definition sketch: (a) physical domain; and (b) transformed domain.

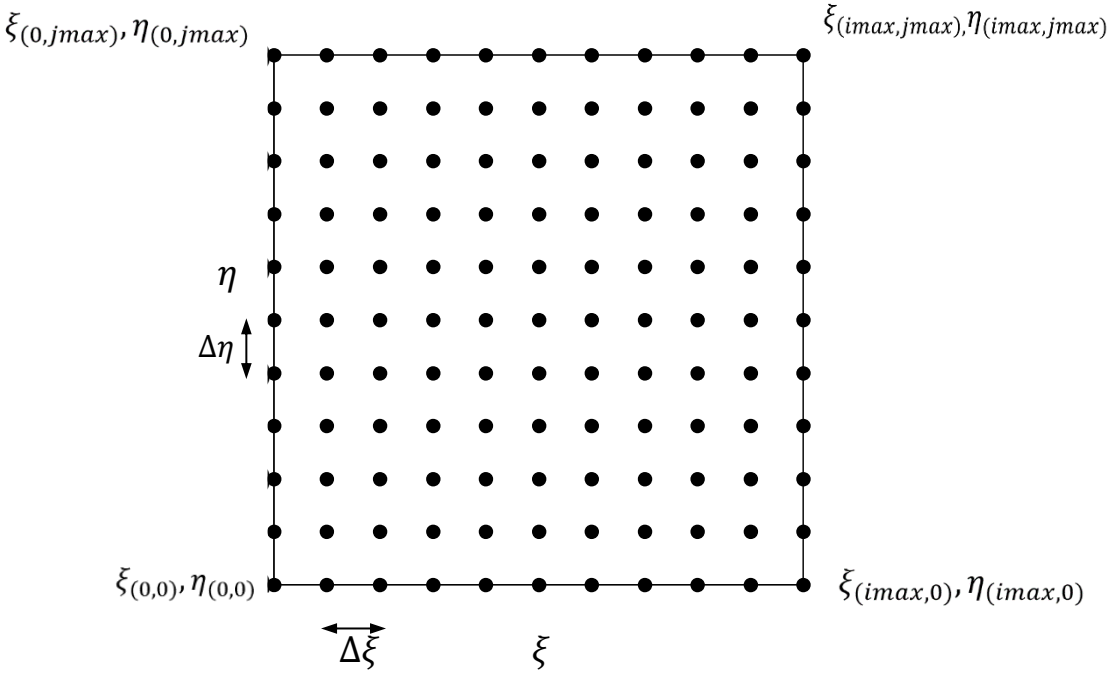


Figure 2.2 Discretised grid in transformed computational domain

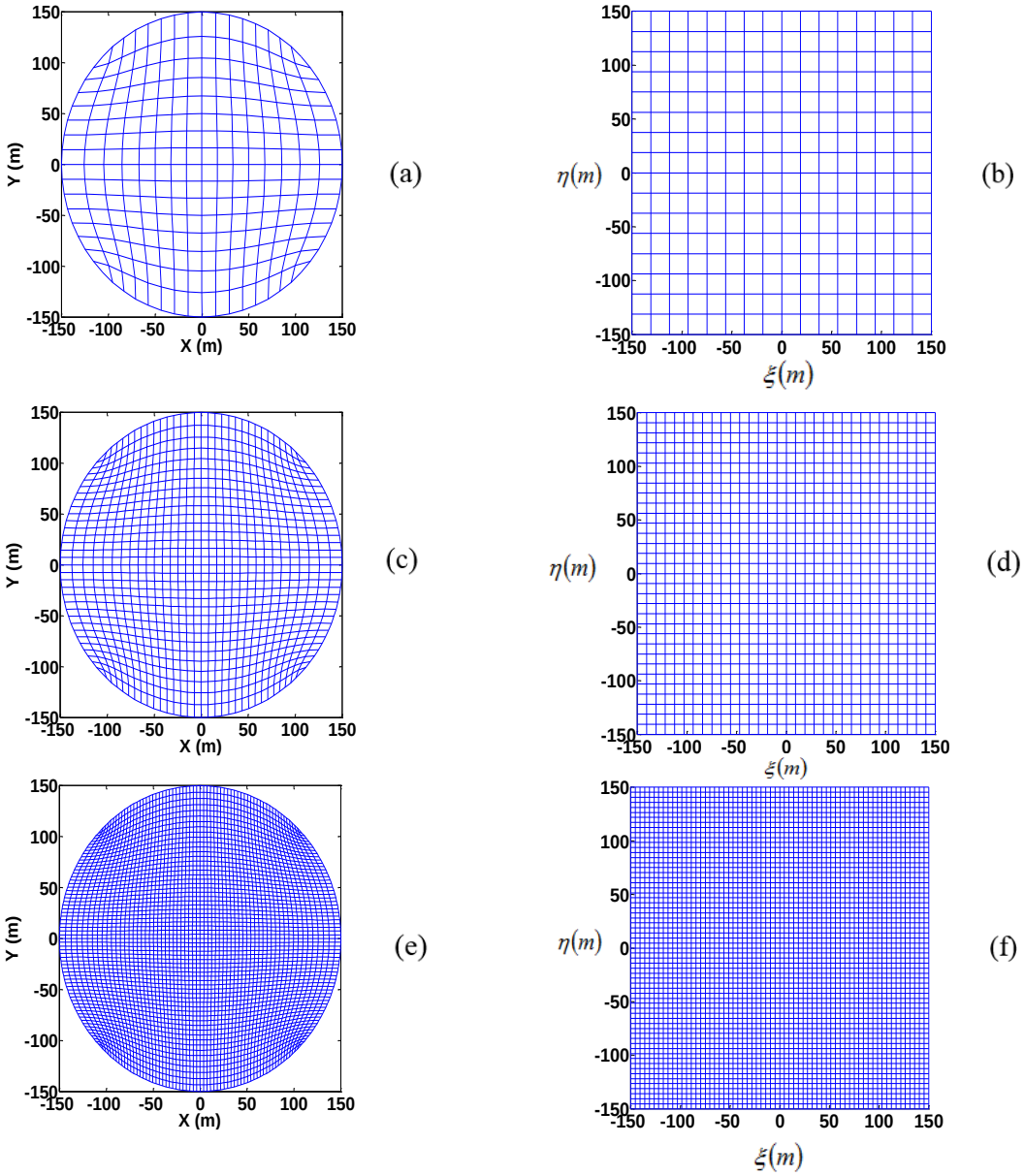


Figure 2.3 Circular grids generated by solving the curvilinear grid generation equations using successive over-relaxation: (a) physical domain (16 x 16 grid); (b) transformed domain (16 x 16 grid); (c) physical domain (32 x 32 grid); (d) transformed domain (32 x 32 grid); (e) physical domain (64 x 64 grid); and (f) transformed domain (64 x 64 grid).

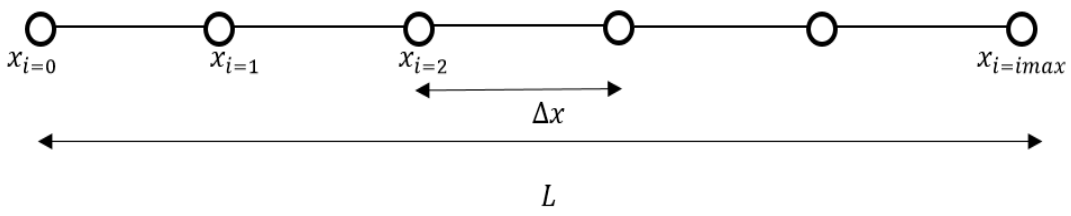


Figure 2.4 Example one-dimensional grid for multi-grid solver

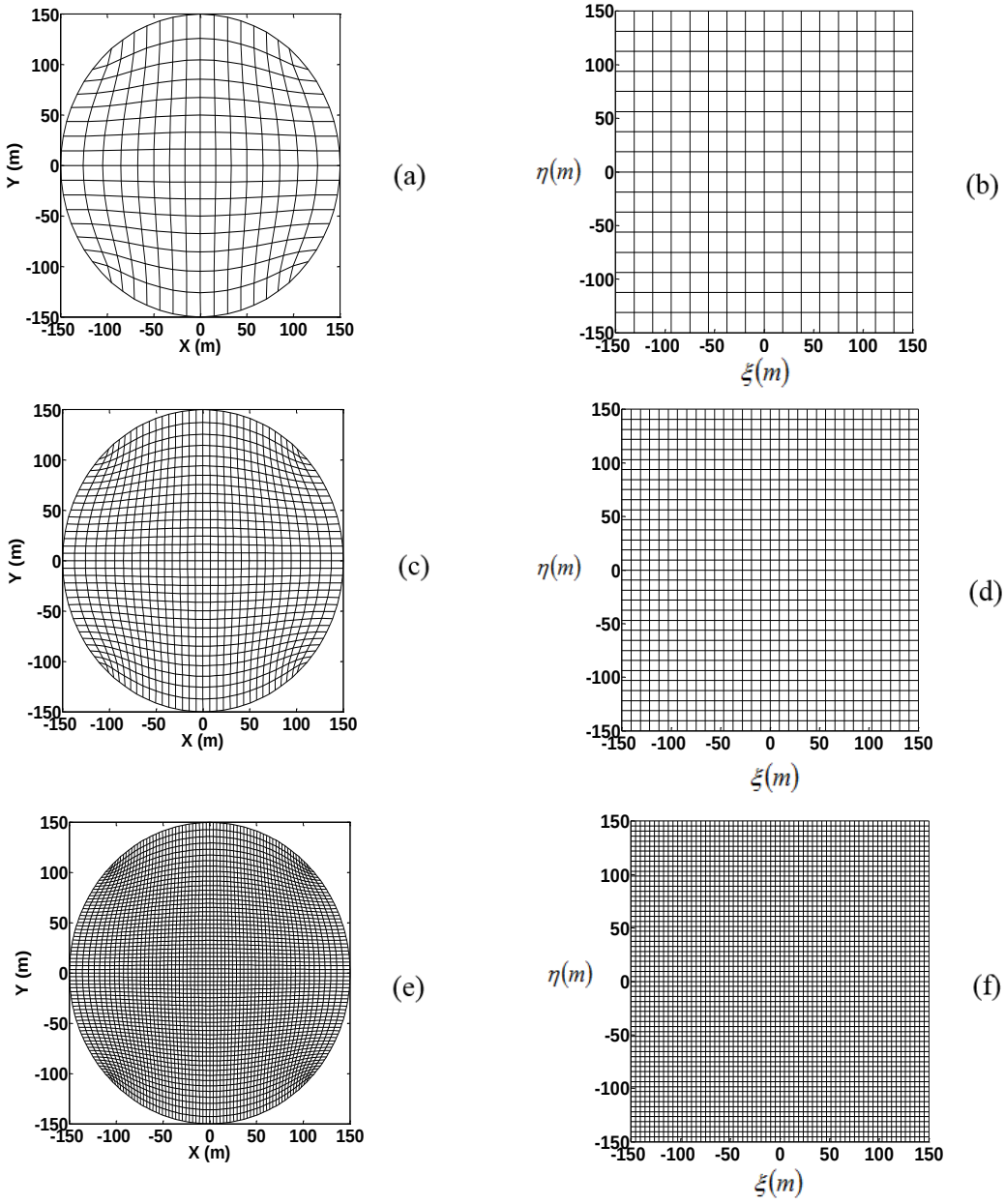


Figure 2.5 Circular grids generated by solving the curvilinear grid generation equations using multi-grid iteration: (a) physical domain (16 x 16 grid); (b) transformed domain (16 x 16 grid); (c) physical domain (32 x 32 grid); (d) transformed domain (32 x 32 grid); (e) physical domain (64 x 64 grid); and (f) transformed domain (64 x 64 grid).

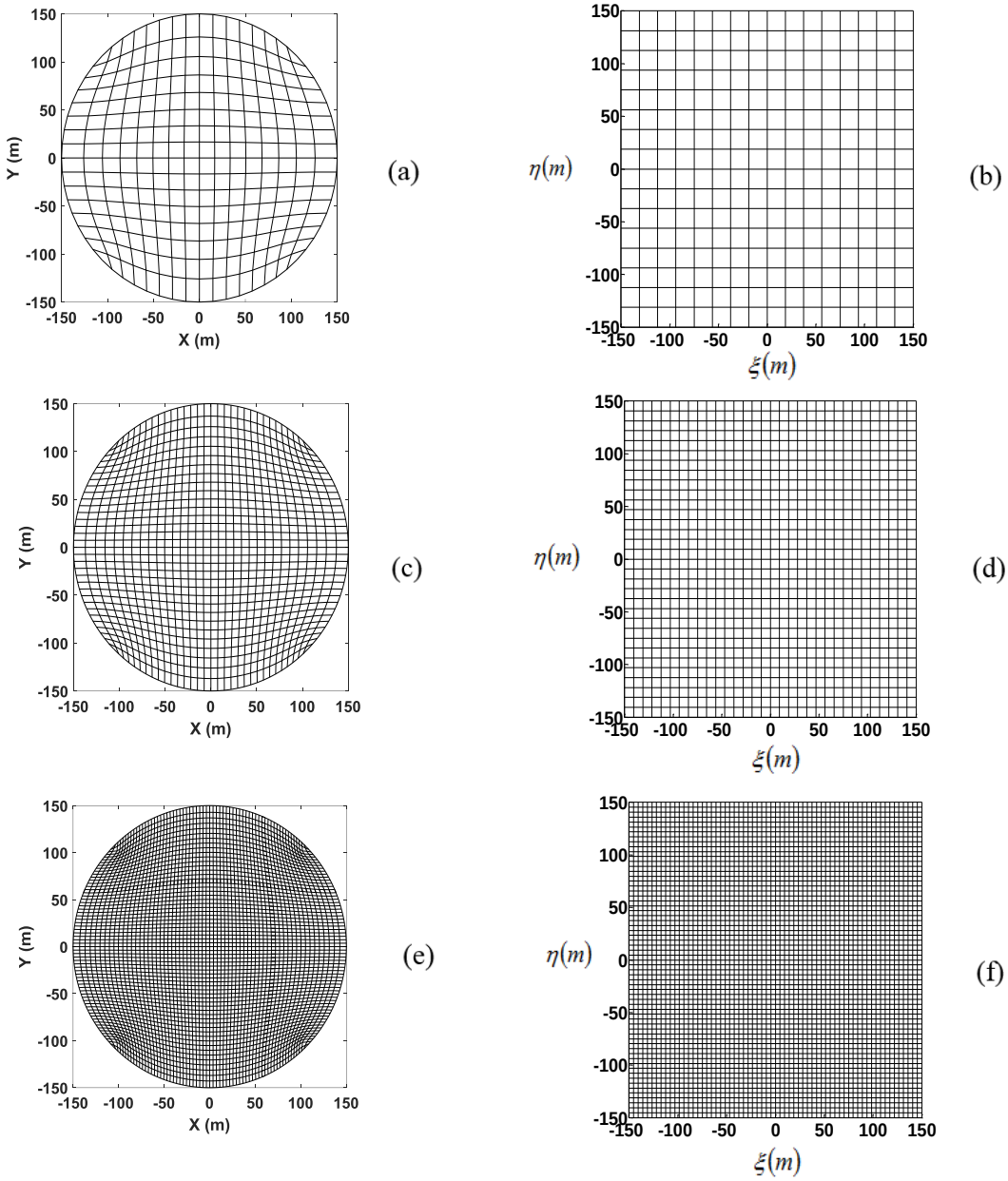


Figure 2.6 Circular grids generated by solving the curvilinear grid generation equations using conjugate gradient method: (a) physical domain (16 x 16 grid); (b) transformed domain (16 x 16 grid); (c) physical domain (32 x 32 grid); (d) transformed domain (32 x 32 grid); (e) physical domain (64 x 64 grid); and (f) transformed domain (64 x 64 grid).

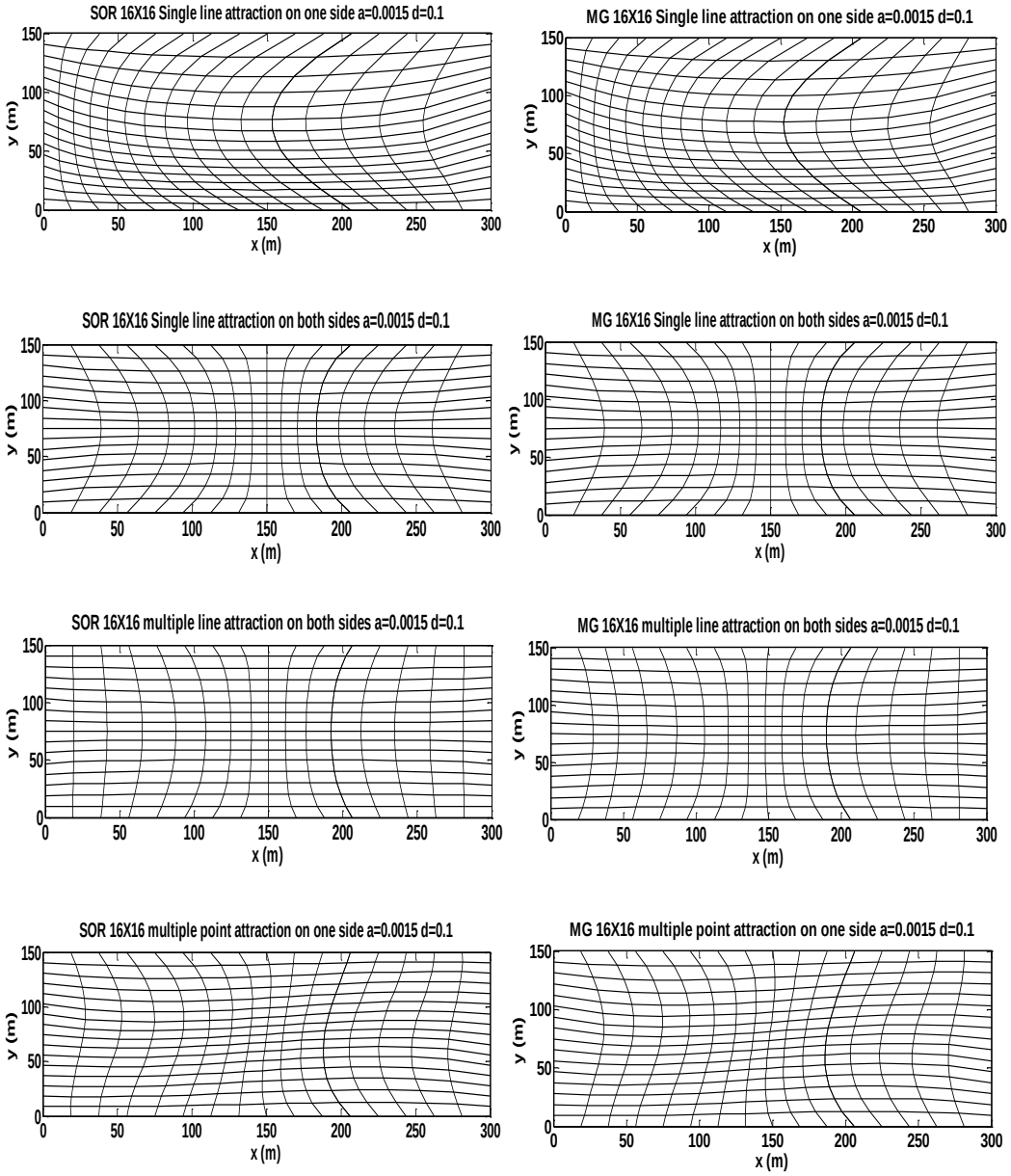


Figure 2.7 Distorted 16 x 16 curvilinear grids generated within a rectangle using successive over-relaxation (SOR) and multi-grid (MG) iteration

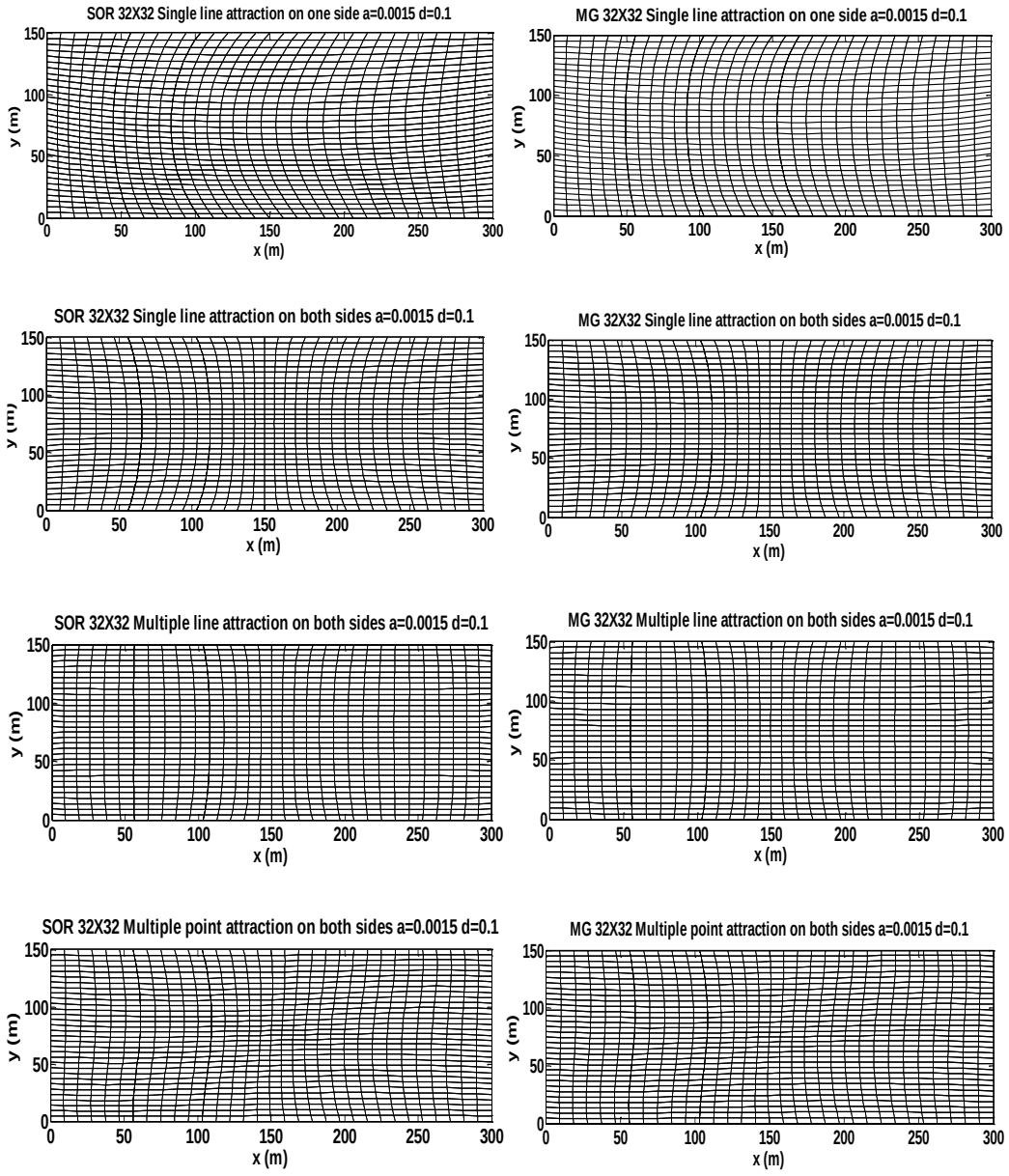


Figure 2.8 Distorted 32 x 32 curvilinear grids generated within a rectangle using successive over-relaxation (SOR) and multi-grid (MG) iteration

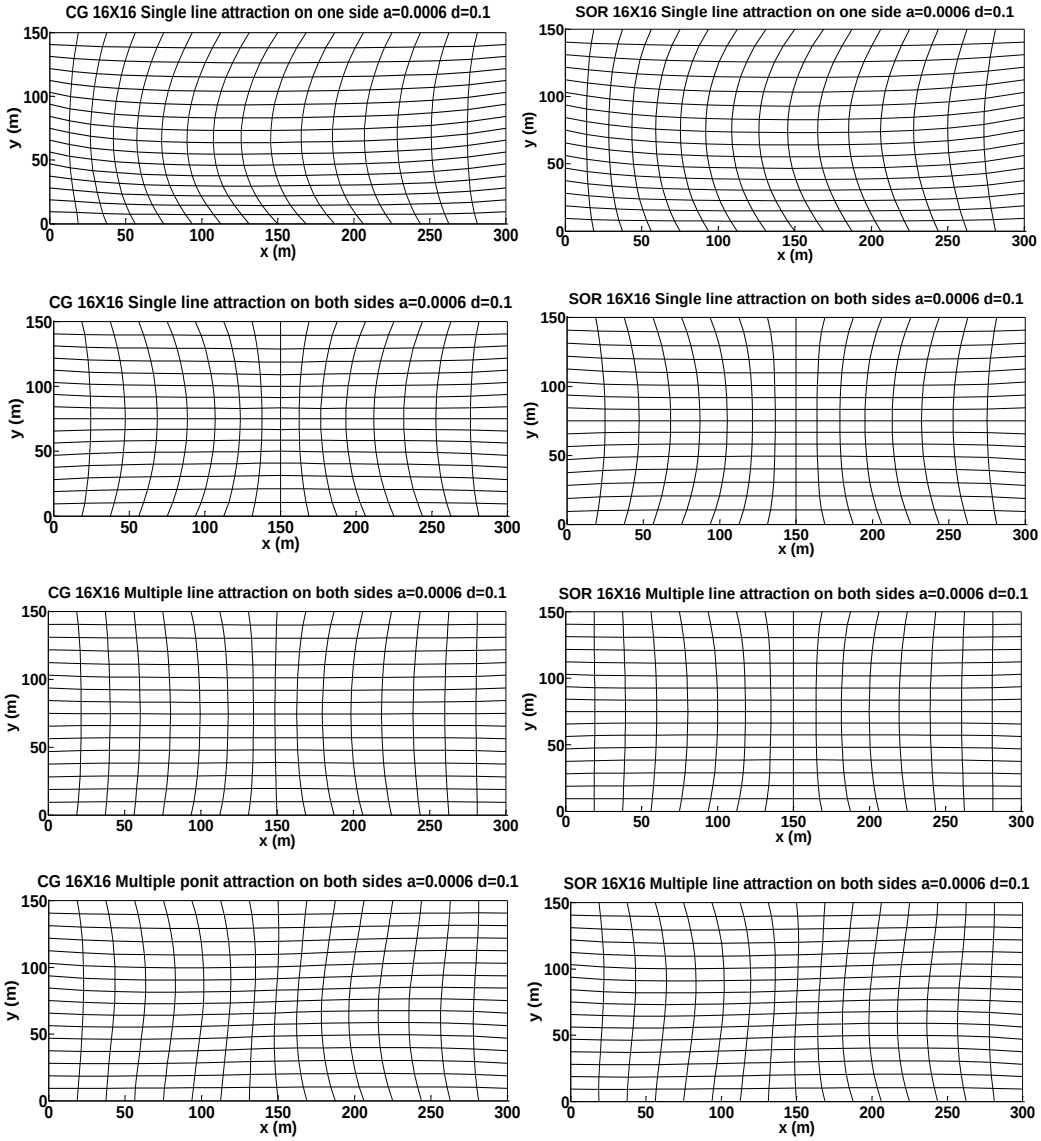


Figure 2.9 Distorted 16 x 16 curvilinear grids generated within a rectangle using conjugate gradient method (CG) and successive over-relaxation (SOR)

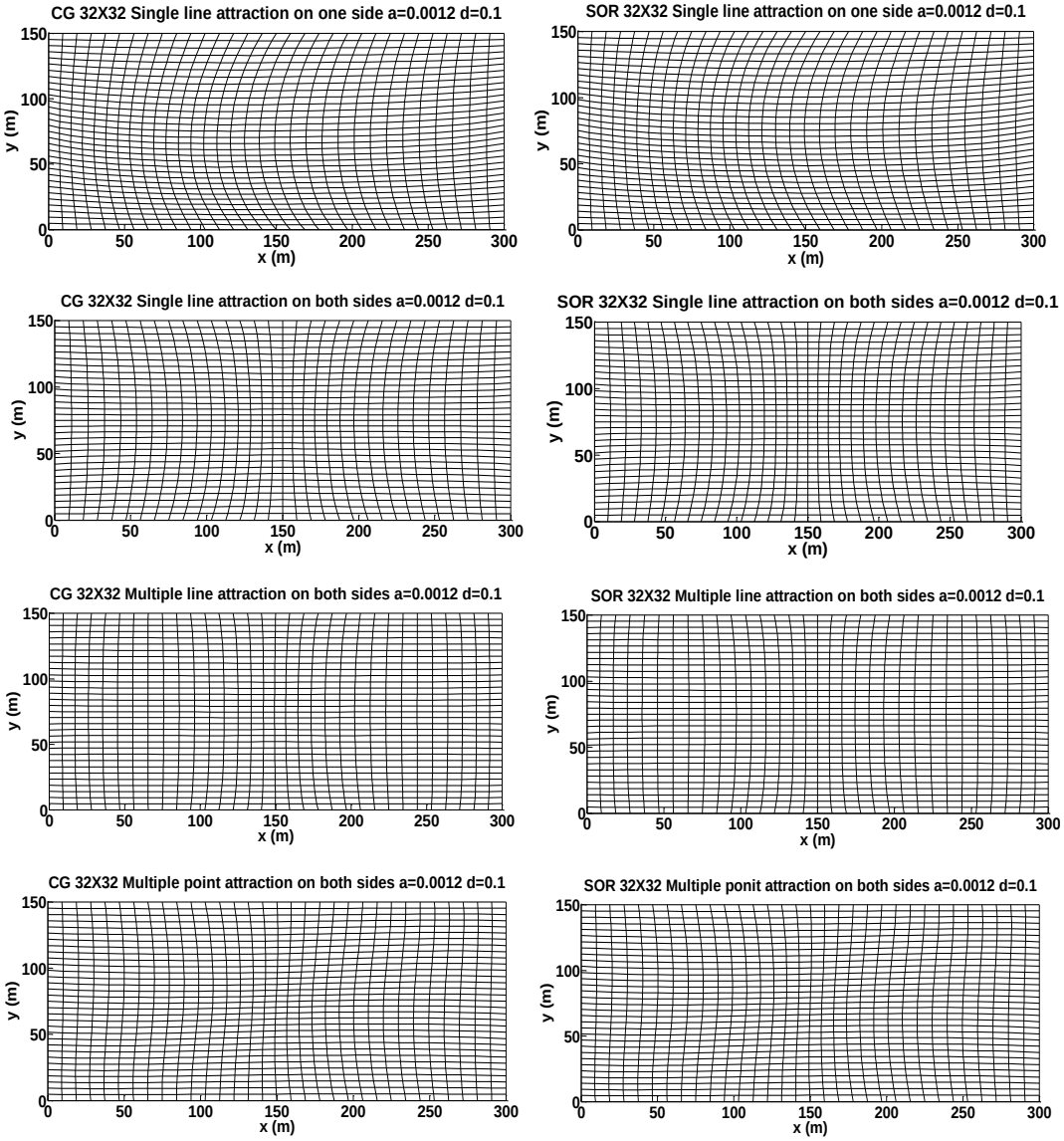


Figure 2.10 Distorted 32 x 32 curvilinear grids generated within a rectangle using conjugate gradient method (CG) and successive over-relaxation (SOR)

Chapter 3

Shallow Water Equations

This chapter presents a detailed derivation of the shallow water equations in both Cartesian and curvilinear forms, a finite difference discretization of the equations, and the procedure used for their numerical solution.

The Shallow Water Equations (SWEs) apply to nearly horizontal flows in wide open channels, rivers, and lakes, and so are at the heart of the work presented in this thesis. Figure 3.1 is a definition sketch, indicating the bed elevation, still water depth, free surface elevation and local total depth in a shallow flow. In deriving the shallow water equations, it is assumed that the hydrodynamics is that of long free surface waves in incompressible shallow water, such that $kh_s < \pi/10$ or $h_s < \lambda/20$ (where h_s is the mean water depth, $k = \frac{2\pi}{\lambda}$ is the wavenumber, and λ is the wavelength). It is also assumed that the waves are small-amplitude waves such that their amplitude. With these assumptions, the vertical acceleration component can be neglected implying hydrostatic pressure. The shallow water equations are usually derived in one of three ways: fundamental control volume analysis of three-dimensional elements and depth-averaging; control volume analysis of elements that extend through the depth with velocity assumed uniform in the vertical; and from potential theory. Here, the SWEs comprise fundamental fluid dynamic equations related to mass conservation (continuity) and force/momentum balance.

To derive the Shallow Water Equations commencing with the fundamental continuity and Navier-Stokes momentum equations in 3-D, the equations are converted into their Reynolds-averaged form, and then integrated over the depth. The first mean value theorem of calculus is used to define depth-averaged velocity components. By introducing suitable bed and free surface boundary conditions, and assuming that vertical acceleration is negligible such that the pressure profile is hydrostatic, the shallow water equations turn out to be much simpler than their 3-D counterparts.

3.1 Continuity and Navier-Stokes momentum equations

The mass balance between fluxes entering and leaving the control volume depicted in Figure 3.2a and the mass accumulated within the element may be expressed

$$\frac{\partial \rho}{\partial t} + \frac{\partial(\rho u)}{\partial x} + \frac{\partial(\rho v)}{\partial y} + \frac{\partial(\rho w)}{\partial z} = 0 \quad , \qquad (3.1)$$

where ρ is the fluid density, u, v and w are the instantaneous fluid velocity components in the Cartesian x, y and z directions, and t is time. For an incompressible fluid, the density is a constant, and this simplifies to give the continuity equation

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z} = 0 \ . \qquad (3.2)$$

Let us consider the force balance across the element shown in Figure 3.2b, which indicates the fluid stresses acting in the x-direction. There are similar stresses acting in the y- and z-directions. By considering the force balance, and applying Newton's second law, the following three expressions for the fluid acceleration components:

$$\begin{aligned}
\frac{\mathrm{d}u}{\mathrm{d}t} &= X + \frac{1}{\rho}\left[\frac{\partial \sigma_{xx}}{\partial x} + \frac{\partial \tau_{yx}}{\partial y} + \frac{\partial \tau_{zx}}{\partial z}\right] \\
\frac{\mathrm{d}v}{\mathrm{d}t} &= Y + \frac{1}{\rho}\left[\frac{\partial \tau_{xy}}{\partial x} + \frac{\partial \sigma_{yy}}{\partial y} + \frac{\partial \tau_{zy}}{\partial z}\right] \qquad (3.3) \\
\frac{\mathrm{d}w}{\mathrm{d}t} &= Z + \frac{1}{\rho}\left[\frac{\partial \tau_{xz}}{\partial x} + \frac{\partial \tau_{yz}}{\partial y} + \frac{\partial \sigma_{zz}}{\partial z}\right] \quad .
\end{aligned}$$

in which where X, Y and Z are the body force components in the x, y, and z directions, σ_{ii} represents the normal stress component to face i, and τ_{ij} represents the shear stress components on face i. The shear stresses obey symmetry so that $\tau_{ij} = \tau_{ji}$. The normal stress components comprise the sum of the mean stress (i.e. pressure) and the deviatoric stress components proportional to the rate of change of strain with time,

such that $\sigma_{xx} = p + \sigma'_{xx}$, $\sigma_{yy} = \bar{\sigma} + \sigma'_{yy}$, and $\sigma_{zz} = \bar{\sigma} + \sigma'_{zz}$ where $\bar{\sigma} = \frac{1}{3}(\sigma_{xx} + \sigma_{yy} + \sigma_{zz}) = -p$. Hence, we can write

$$\begin{aligned}
\frac{\mathrm{d}u}{\mathrm{d}t} &= X - \frac{1}{\rho}\frac{\partial p}{\partial x} + \frac{1}{\rho}\left[\frac{\partial \sigma'_{xx}}{\partial x} + \frac{\partial \tau_{yx}}{\partial y} + \frac{\partial \tau_{zx}}{\partial z}\right] \\
\frac{\mathrm{d}v}{\mathrm{d}t} &= Y - \frac{1}{\rho}\frac{\partial p}{\partial y} + \frac{1}{\rho}\left[\frac{\partial \tau_{xy}}{\partial x} + \frac{\partial \sigma'_{yy}}{\partial y} + \frac{\partial \tau_{zy}}{\partial z}\right] \qquad (3.4) \\
\frac{\mathrm{d}w}{\mathrm{d}t} &= Z - \frac{1}{\rho}\frac{\partial p}{\partial z} + \frac{1}{\rho}\left[\frac{\partial \tau_{xz}}{\partial x} + \frac{\partial \tau_{yz}}{\partial y} + \frac{\partial \sigma'_{zz}}{\partial z}\right] .
\end{aligned}$$

Using Stokes' (1845) hypothesis, the six stress components are related to the corresponding six rates of strain for an **isotropic** (where the relationships between stress and rate of strain are independent of direction), **Newtonian** (where the shear stress is linearly proportional to the rate of strain) **fluid** by the following constitutive equations:

$$\begin{aligned}
\sigma_{xx} &= -p + \sigma'_{xx} = -p + 2\mu\frac{\partial u}{\partial x} - \frac{2}{3}\mu\left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z}\right) \\
\sigma_{yy} &= -p + \sigma'_{yy} = -p + 2\mu\frac{\partial v}{\partial y} - \frac{2}{3}\mu\left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z}\right) \qquad (3.5) \\
\sigma_{zz} &= -p + \sigma'_{zz} = -p + 2\mu\frac{\partial w}{\partial z} - \frac{2}{3}\mu\left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z}\right) \\
\tau_{xy} &= \tau_{yx} = \mu\left(\frac{\partial v}{\partial x} + \frac{\partial u}{\partial y}\right) \\
\tau_{xz} &= \tau_{zx} = \mu\left(\frac{\partial w}{\partial x} + \frac{\partial u}{\partial z}\right) \\
\tau_{yz} &= \tau_{zy} = \mu\left(\frac{\partial w}{\partial y} + \frac{\partial v}{\partial z}\right) .
\end{aligned}$$

in which μ is the dynamic viscosity of the fluid.

Substituting the above formulae into the momentum equations gives the non-conservative form of the Navier-Stokes momentum equations for incompressible flow which may be written as follows:

$$\begin{aligned}
\frac{\mathrm{d}u}{\mathrm{d}t} &= X - \frac{1}{\rho}\frac{\partial p}{\partial x} + \nu\left[\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 u}{\partial z^2}\right] \\
\frac{\mathrm{d}v}{\mathrm{d}t} &= Y - \frac{1}{\rho}\frac{\partial p}{\partial y} + \nu\left[\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} + \frac{\partial^2 v}{\partial z^2}\right] \\
\frac{\mathrm{d}w}{\mathrm{d}t} &= Z - \frac{1}{\rho}\frac{\partial p}{\partial z} + \nu\left[\frac{\partial^2 w}{\partial x^2} + \frac{\partial^2 w}{\partial y^2} + \frac{\partial^2 w}{\partial z^2}\right]
\end{aligned} \tag{3.6}$$

By definition,

$$\begin{aligned}
\frac{\mathrm{d}u}{\mathrm{d}t} &= \frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} + w\frac{\partial u}{\partial z} \\
\frac{\mathrm{d}v}{\mathrm{d}t} &= \frac{\partial v}{\partial t} + u\frac{\partial v}{\partial x} + v\frac{\partial v}{\partial y} + w\frac{\partial v}{\partial z} \\
\frac{\mathrm{d}w}{\mathrm{d}t} &= \frac{\partial w}{\partial t} + u\frac{\partial w}{\partial x} + v\frac{\partial w}{\partial y} + w\frac{\partial w}{\partial z}
\end{aligned} \tag{3.7}$$

Hence, the non-conservative Navier-Stokes equations are

$$\begin{aligned}
\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} + w\frac{\partial u}{\partial z} &= X - \frac{1}{\rho}\frac{\partial p}{\partial x} + \nu\left[\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 u}{\partial z^2}\right] \\
\frac{\partial v}{\partial t} + u\frac{\partial v}{\partial x} + v\frac{\partial v}{\partial y} + w\frac{\partial v}{\partial z} &= Y - \frac{1}{\rho}\frac{\partial p}{\partial y} + \nu\left[\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} + \frac{\partial^2 v}{\partial z^2}\right] \\
\frac{\partial w}{\partial t} + u\frac{\partial w}{\partial x} + v\frac{\partial w}{\partial y} + w\frac{\partial w}{\partial z} &= Z - \frac{1}{\rho}\frac{\partial p}{\partial z} + \nu\left[\frac{\partial^2 w}{\partial x^2} + \frac{\partial^2 w}{\partial y^2} + \frac{\partial^2 w}{\partial z^2}\right].
\end{aligned} \tag{3.8}$$

By adding the continuity equation multiplied by each of the velocity components to the corresponding momentum equation, the non-conservative form of the Navier-Stokes momentum equations is obtained as follows:

$$\frac{\partial u}{\partial t}+\frac{\partial u^2}{\partial x}+\frac{\partial uv}{\partial y}+\frac{\partial uw}{\partial z}=X-\frac{1}{\rho}\frac{\partial p}{\partial x}+\nu\left[\frac{\partial^2 u}{\partial x^2}+\frac{\partial^2 u}{\partial y^2}+\frac{\partial^2 u}{\partial z^2}\right]$$

$$\frac{\partial v}{\partial t}+\frac{\partial uv}{\partial x}+\frac{\partial v^2}{\partial y}+\frac{\partial vw}{\partial z}=Y-\frac{1}{\rho}\frac{\partial p}{\partial y}+\nu\left[\frac{\partial^2 v}{\partial x^2}+\frac{\partial^2 v}{\partial y^2}+\frac{\partial^2 v}{\partial z^2}\right] \qquad (3.9)$$

$$\frac{\partial w}{\partial t}+\frac{\partial uw}{\partial x}+\frac{\partial vw}{\partial y}+\frac{\partial w^2}{\partial z}=Z-\frac{1}{\rho}\frac{\partial p}{\partial z}+\nu\left[\frac{\partial^2 w}{\partial x^2}+\frac{\partial^2 w}{\partial y^2}+\frac{\partial^2 w}{\partial z^2}\right].$$

3.2 Reynolds-averaged Continuity and Navier-Stokes equations:

Almost all environmental flows are turbulent, whereby irregular fluctuations arising from mixing and eddying motions are superimposed on the main stream. In rivers, large eddies develop in the bed boundary layer or as the flow passes bluff obstacles (such as bridge piers) and these eddies extract energy continuously from the main flow. Small eddies dissipate the energy, which is finally removed by viscosity. Following Osborne Reynolds (1895) it is usual to average the continuity and Navier-Stokes equations over relatively small time intervals in order to handle turbulent fluctuations due to the smallest eddies, and simplify the problem. To implement Reynolds-averaging, we first split the instantaneous velocity components and pressure into the sum of time-averaged and fluctuating values as follows:

$$\begin{aligned} u &= \bar{u}+u', \\ v &= \bar{v}+v', \\ w &= \bar{w}+w', \\ \text{and} \quad p &= \bar{p}+p'. \end{aligned} \qquad (3.10)$$

Let the variable *f* represent either a velocity component or pressure component. Then, $f=\bar{f}+f'$ where $\bar{f}$ is the mean component and f' is the fluctuating component. By definition, $\bar{f}=\frac{1}{\Delta t}\int_t^{t+\Delta t} f\ dt$ and $\int_t^{t+\Delta t} f'\ dt=0$. After time-averaging the continuity equation and the Navier-Stokes equations over a sufficient length of time for the mean values to be independent of time we obtain the conservative form of the Reynolds equations

$$\frac{\partial \overline{u}}{\partial x}+\frac{\partial \overline{v}}{\partial y}+\frac{\partial \overline{w}}{\partial z}=0\,, \tag{3.11}$$

$$\begin{aligned}
&\frac{\partial \overline{u}}{\partial t}+\frac{\partial \overline{u}^{2}}{\partial x}+\frac{\partial \overline{u}\,\overline{v}}{\partial y}+\frac{\partial \overline{u}\,\overline{w}}{\partial z}=\overline{X}-\frac{1}{\rho}\frac{\partial \overline{p}}{\partial x}+\nu\left[\frac{\partial^{2}\overline{u}}{\partial x^{2}}+\frac{\partial^{2}\overline{u}}{\partial y^{2}}+\frac{\partial^{2}\overline{u}}{\partial z^{2}}\right]-\left[\frac{\partial \overline{u'u'}}{\partial x}+\frac{\partial \overline{u'v'}}{\partial y}+\frac{\partial \overline{u'w'}}{\partial z}\right]\\
&\frac{\partial \overline{v}}{\partial t}+\frac{\partial \overline{u}\,\overline{v}}{\partial x}+\frac{\partial \overline{v}^{2}}{\partial y}+\frac{\partial \overline{v}\,\overline{w}}{\partial z}=\overline{Y}-\frac{1}{\rho}\frac{\partial \overline{p}}{\partial y}+\nu\left[\frac{\partial^{2}\overline{v}}{\partial x^{2}}+\frac{\partial^{2}\overline{v}}{\partial y^{2}}+\frac{\partial^{2}\overline{v}}{\partial z^{2}}\right]-\left[\frac{\partial \overline{v'u'}}{\partial x}+\frac{\partial \overline{v'v'}}{\partial y}+\frac{\partial \overline{v'w'}}{\partial z}\right]\\
&\frac{\partial \overline{w}}{\partial t}+\frac{\partial \overline{u}\,\overline{w}}{\partial x}+\frac{\partial \overline{v}\,\overline{w}}{\partial y}+\frac{\partial \overline{w}^{2}}{\partial z}=\overline{Z}-\frac{1}{\rho}\frac{\partial \overline{p}}{\partial z}+\nu\left[\frac{\partial^{2}\overline{w}}{\partial x^{2}}+\frac{\partial^{2}\overline{w}}{\partial y^{2}}+\frac{\partial^{2}\overline{w}}{\partial z^{2}}\right]-\left[\frac{\partial \overline{w'u'}}{\partial x}+\frac{\partial \overline{w'v'}}{\partial y}+\frac{\partial \overline{w'w'}}{\partial z}\right].
\end{aligned} \tag{3.12}$$

Again, by subtracting the Reynolds-averaged continuity equation multiplied by each time-averaged velocity component from the corresponding Reynolds-averaged momentum equation, the non-conservative form of the Reynolds-averaged Navier-Stokes equations is obtained as:

$$\begin{aligned}
&\frac{\partial \overline{u}}{\partial t}+\overline{u}\frac{\partial \overline{u}}{\partial x}+\overline{v}\frac{\partial \overline{u}}{\partial y}+\overline{w}\frac{\partial \overline{u}}{\partial z}=\overline{X}-\frac{1}{\rho}\frac{\partial \overline{p}}{\partial x}+\\
&+\frac{1}{\rho}\left[\frac{\partial}{\partial x}\left(\mu\frac{\partial \overline{u}}{\partial x}-\rho\overline{u'u'}\right)+\frac{\partial}{\partial y}\left(\mu\frac{\partial \overline{u}}{\partial y}-\rho\overline{u'v'}\right)+\frac{\partial}{\partial z}\left(\mu\frac{\partial \overline{u}}{\partial z}-\rho\overline{u'w'}\right)\right]\\
&\frac{\partial \overline{v}}{\partial t}+\overline{u}\frac{\partial \overline{v}}{\partial x}+\overline{v}\frac{\partial \overline{v}}{\partial y}+\overline{w}\frac{\partial \overline{v}}{\partial z}=\overline{Y}-\frac{1}{\rho}\frac{\partial \overline{p}}{\partial y}+\\
&+\frac{1}{\rho}\left[\frac{\partial}{\partial x}\left(\mu\frac{\partial \overline{v}}{\partial x}-\rho\overline{v'u'}\right)+\frac{\partial}{\partial y}\left(\mu\frac{\partial \overline{v}}{\partial y}-\rho\overline{v'v'}\right)+\frac{\partial}{\partial z}\left(\mu\frac{\partial \overline{v}}{\partial z}-\rho\overline{v'w'}\right)\right]\\
&\frac{\partial \overline{w}}{\partial t}+\overline{u}\frac{\partial \overline{w}}{\partial x}+\overline{v}\frac{\partial \overline{w}}{\partial y}+\overline{w}\frac{\partial \overline{w}}{\partial z}=\overline{Z}-\frac{1}{\rho}\frac{\partial \overline{p}}{\partial z}+\\
&+\frac{1}{\rho}\left[\frac{\partial}{\partial x}\left(\mu\frac{\partial \overline{w}}{\partial x}-\rho\overline{w'u'}\right)+\frac{\partial}{\partial y}\left(\mu\frac{\partial \overline{w}}{\partial y}-\rho\overline{w'v'}\right)+\frac{\partial}{\partial z}\left(\mu\frac{\partial \overline{w}}{\partial z}-\rho\overline{w'w'}\right)\right]
\end{aligned} \tag{3.13}$$

The Reynolds-averaged Navier-Stokes equations are effectively the Navier-Stokes equations with time-averaged quantities replacing the instantaneous quantities and the addition of Reynolds stress terms that deal with correlations between the fluctuating turbulent quantities.

The **Reynolds (apparent) stresses** are:

$$-\rho\overline{u'u'},\quad -\rho\overline{u'v'},\quad -\rho\overline{u'w'},\quad -\rho\overline{v'v'},\quad -\rho\overline{v'w'},\quad -\rho\overline{w'w'}.$$

Using the Boussinesq approximation, the Reynolds stresses may be related to diffusion processes through a mixing coefficient, such that

$$-\rho\overline{u'u'}=\eta\left(\frac{\partial\overline{u}}{\partial x}+\frac{\partial\overline{u}}{\partial x}\right),\quad -\rho\overline{u'v'}=\eta\left(\frac{\partial\overline{u}}{\partial y}+\frac{\partial\overline{v}}{\partial x}\right),\quad -\rho\overline{u'w'}=\eta\left(\frac{\partial\overline{u}}{\partial z}+\frac{\partial\overline{w}}{\partial x}\right),$$

$$-\rho\overline{v'v'}=\eta\left(\frac{\partial\overline{v}}{\partial y}+\frac{\partial\overline{v}}{\partial y}\right),\quad -\rho\overline{v'w'}=\eta\left(\frac{\partial\overline{v}}{\partial z}+\frac{\partial\overline{w}}{\partial y}\right),\quad -\rho\overline{w'w'}=\eta\left(\frac{\partial\overline{w}}{\partial z}+\frac{\partial\overline{w}}{\partial z}\right).$$

where η is the absolute eddy viscosity. Note that the kinematic eddy viscosity is defined as $\varepsilon=\frac{\eta}{\rho}$, and that the turbulent viscosities are usually several orders of magnitude larger than their laminar counterparts, such that $\eta>>\mu$ and $\varepsilon>>\nu$.

3.3 Depth-integrated Reynolds-averaged Equations

The shallow water equations may now be derived by depth-integration of the Reynolds-averaged equations, while applying suitable boundary conditions at the bed and free surface. Figure 3.1 is a definition sketch, illustrating the key geometric variables; here, the total depth is given by $h=h_s+\zeta$ where h_s is the still water depth, ζ is the free surface elevation above still water level and z_b is the distance between the bed and the horizontal datum.

From before, the Reynolds-averaged continuity and momentum equations may be written

$$\frac{\partial\overline{u}}{\partial x}+\frac{\partial\overline{v}}{\partial y}+\frac{\partial\overline{w}}{\partial z}=0\quad ,$$

$$\frac{\partial\overline{u}}{\partial t}+\frac{\partial\overline{u}^2}{\partial x}+\frac{\partial\overline{u}\overline{v}}{\partial y}+\frac{\partial\overline{u}\overline{w}}{\partial z}=\overline{X}-\frac{1}{\rho}\frac{\partial\overline{p}}{\partial x}+\frac{1}{\rho}\left[\frac{\partial\sigma'_{xx}}{\partial x}+\frac{\partial\tau_{xy}}{\partial y}+\frac{\partial\tau_{xz}}{\partial z}\right]$$

$$\frac{\partial \overline{v}}{\partial t}+\frac{\partial \overline{u}\,\overline{v}}{\partial x}+\frac{\partial \overline{v}^{2}}{\partial y}+\frac{\partial \overline{v}\,\overline{w}}{\partial z}=\overline{Y}-\frac{1}{\rho}\frac{\partial \overline{p}}{\partial y}+\frac{1}{\rho}\left[\frac{\partial \tau_{xy}}{\partial x}+\frac{\partial \sigma'_{yy}}{\partial y}+\frac{\partial \tau_{zy}}{\partial z}\right]$$

$$\frac{\partial \overline{w}}{\partial t}+\frac{\partial \overline{u}\,\overline{w}}{\partial x}+\frac{\partial \overline{v}\,\overline{w}}{\partial y}+\frac{\partial \overline{w}^{2}}{\partial z}=\overline{Z}-\frac{1}{\rho}\frac{\partial \overline{p}}{\partial z}+\frac{1}{\rho}\left[\frac{\partial \tau_{xz}}{\partial x}+\frac{\partial \tau_{yz}}{\partial y}+\frac{\partial \sigma'_{zz}}{\partial z}\right].$$

in which

$$\sigma'_{xx}=2\mu\frac{\partial \overline{u}}{\partial x}-\rho\overline{u'u'},\qquad \sigma'_{yy}=2\mu\frac{\partial \overline{v}}{\partial y}-\rho\overline{v'v'},\qquad \sigma'_{zz}=2\mu\frac{\partial \overline{w}}{\partial z}-\rho\overline{w'w'}$$

$$\tau_{xy}=\tau_{yx}=\mu\left(\frac{\partial \overline{u}}{\partial y}+\frac{\partial \overline{v}}{\partial x}\right)-\rho\overline{u'v'},\qquad \tau_{xz}=\tau_{zx}=\mu\left(\frac{\partial \overline{u}}{\partial z}+\frac{\partial \overline{w}}{\partial x}\right)-\rho\overline{u'w'}$$

and

$$\tau_{yz}=\tau_{xy}=\mu\left(\frac{\partial \overline{v}}{\partial z}+\frac{\partial \overline{w}}{\partial y}\right)-\rho\overline{v'w'}.$$

In order to carry out depth-integration of the Reynolds-averaged continuity and momentum equations, boundary conditions are required at the bed and free surface. For a solid, but possibly erodible bed, the kinematic boundary condition is one of no-flow, such that

$$\overline{w}\big|_{z=-h_s}=\left.\frac{\mathrm{d}(-h_s)}{\mathrm{d}t}\right|_{-h_s}=\left.\frac{\partial(-h_s)}{\partial t}\right|_{-h_s}+\overline{u}\big|_{-h_s}\left.\frac{\partial(-h_s)}{\partial x}\right|_{-h_s}+\overline{v}\big|_{-h_s}\left.\frac{\partial(-h_s)}{\partial y}\right|_{-h_s}. \qquad (3.14)$$

At the free surface, the kinematic boundary condition stipulates that particles on the free surface remain there (in other words, the total rate of change of the free surface is equal to the vertical velocity of the free surface, such that

$$\overline{w}\big|_{z=\zeta}=\left.\frac{\mathrm{d}\zeta}{\mathrm{d}t}\right|_{\zeta}=\left.\frac{\partial\zeta}{\partial t}\right|_{\zeta}+\overline{u}\big|_{\zeta}\left.\frac{\partial\zeta}{\partial x}\right|_{\zeta}+\overline{v}\big|_{\zeta}\left.\frac{\partial\zeta}{\partial y}\right|_{\zeta}. \qquad (3.15)$$

The continuity equation is depth-integrated from the bed at $z=-h_s$ to the free surface at $z=\zeta$, such that

$$\int_{-h_s}^{\zeta}\frac{\partial \overline{u}}{\partial x}\mathrm{d}z+\int_{-h_s}^{\zeta}\frac{\partial \overline{v}}{\partial y}\mathrm{d}z+\int_{-h_s}^{\zeta}\frac{\partial \overline{w}}{\partial z}\mathrm{d}z=0,$$

or

$$\int_{-h_s}^{\zeta} \frac{\partial \bar{u}}{\partial x} \mathrm{d}z + \int_{-h_s}^{\zeta} \frac{\partial \bar{v}}{\partial y} \mathrm{d}z + \bar{w}\big|_{\zeta} - \bar{w}\big|_{-h_s} = 0 \,. \tag{3.16}$$

Introducing the depth-averaged velocity components (satisfying the first mean value theorem of integration),

$$U = \frac{1}{h} \int_{-h_s}^{\zeta} \bar{u} \, \mathrm{d}z \quad \text{and} \quad V = \frac{1}{h} \int_{-h_s}^{\zeta} \bar{v} \, \mathrm{d}z \quad . \tag{3.17}$$

and applying Leibnitz's formula for differentiation of an integral,

$$\int_a^b \frac{\partial f(x,y)}{\partial y} \mathrm{d}z = \frac{\partial}{\partial y} \int_a^b f(x,y) \mathrm{d}z - f(b,y) \frac{\partial b}{\partial y} + f(a,y) \frac{\partial a}{\partial y} \,, \tag{3.18}$$

the mass conservation shallow water equation is obtained (after substituting the free surface and bed kinematic boundary conditions) as

$$\frac{\partial \zeta}{\partial t} + \frac{\partial (Uh)}{\partial x} + \frac{\partial (Vh)}{\partial y} = 0 \,. \tag{3.19}$$

Turning to the x and y momentum equations, similar depth-integrations are performed (after substituting in for the body forces, and where the Coriolis parameter is $f = 2v\omega \sin\phi$), such that

$$\int_{-h_s}^{\zeta} \frac{\partial \bar{u}}{\partial t} \mathrm{d}z + \int_{-h_s}^{\zeta} \frac{\partial \bar{u}^2}{\partial x} \mathrm{d}z + \int_{-h_s}^{\zeta} \frac{\partial \bar{u}\bar{v}}{\partial y} \mathrm{d}z + \int_{-h_s}^{\zeta} \frac{\partial \bar{u}\bar{w}}{\partial z} \mathrm{d}z = \int_{-h_s}^{\zeta} f \bar{v} \mathrm{d}z - \frac{1}{\rho} \int_{-h_s}^{\zeta} \frac{\partial \bar{p}}{\partial x} \mathrm{d}z + \\ + \frac{1}{\rho} \int_{-h_s}^{\zeta} \frac{\partial \sigma'_{xx}}{\partial x} \mathrm{d}z + \frac{1}{\rho} \int_{-h_s}^{\zeta} \frac{\partial \tau_{yx}}{\partial y} \mathrm{d}z + \frac{1}{\rho} \int_{-h_s}^{\zeta} \frac{\partial \tau_{zx}}{\partial z} \mathrm{d}z$$

and

$$\int_{-h_s}^{\zeta} \frac{\partial \bar{v}}{\partial t} \mathrm{d}z + \int_{-h_s}^{\zeta} \frac{\partial \bar{u}v}{\partial x} \mathrm{d}z + \int_{-h_s}^{\zeta} \frac{\partial \bar{v}^2}{\partial y} \mathrm{d}z + \int_{-h_s}^{\zeta} \frac{\partial \bar{v}\bar{w}}{\partial z} \mathrm{d}z = -\int_{-h_s}^{\zeta} f \bar{u} \, \mathrm{d}z - \frac{1}{\rho} \int_{-h_s}^{\zeta} \frac{\partial \bar{p}}{\partial y} \mathrm{d}z + \\ + \frac{1}{\rho} \int_{-h_s}^{\zeta} \frac{\partial \tau_{xy}}{\partial x} \mathrm{d}z + \frac{1}{\rho} \int_{-h_s}^{\zeta} \frac{\partial \sigma'_{yy}}{\partial y} \mathrm{d}z + \frac{1}{\rho} \int_{-h_s}^{\zeta} \frac{\partial \tau_{zy}}{\partial z} \mathrm{d}z \tag{3.20}$$

Use of the Leibnitz rule and the kinematic bed and free surface boundary conditions gives (after some manipulation),

$$\frac{\partial(Uh)}{\partial t}+\frac{\partial(U^2h)}{\partial x}+\frac{\partial(UVh)}{\partial y}=fVh-\frac{1}{\rho}\int_{-h_s}^{\zeta}\frac{\partial \bar{p}}{\partial x}\mathrm{d}z+\frac{\tau_{wx}-\tau_{bx}}{\rho}$$
$$+\frac{1}{\rho}\frac{\partial}{\partial x}\int_{-h_x}^{\zeta}\left(2\mu\frac{\partial\bar{u}}{\partial x}-\rho\overline{u'u'}-\rho(\bar{u}-U)^2\right)\mathrm{d}z$$
$$+\frac{1}{\rho}\frac{\partial}{\partial y}\int_{-h_x}^{\zeta}\left(\mu\left(\frac{\partial\bar{u}}{\partial y}+\frac{\partial\bar{v}}{\partial x}\right)-\rho\overline{u'v'}-\rho(\bar{u}-U)(\bar{v}-V)\right)\mathrm{d}z$$

and

$$\frac{\partial(Vh)}{\partial t}+\frac{\partial(UVh)}{\partial x}+\frac{\partial(V^2h)}{\partial y}=-fUh-\frac{1}{\rho}\int_{-h_s}^{\zeta}\frac{\partial \bar{p}}{\partial y}\mathrm{d}z+\frac{\tau_{wy}-\tau_{by}}{\rho}$$
$$+\frac{1}{\rho}\frac{\partial}{\partial x}\int_{-h_x}^{\zeta}\left(\mu\left(\frac{\partial\bar{u}}{\partial y}+\frac{\partial\bar{v}}{\partial x}\right)-\rho\overline{u'v'}-\rho(\bar{u}-U)(\bar{v}-V)\right)\mathrm{d}z \quad (3.21)$$
$$+\frac{1}{\rho}\frac{\partial}{\partial y}\int_{-h_x}^{\zeta}\left(2\mu\frac{\partial\bar{v}}{\partial y}-\rho\overline{v'v'}-\rho(\bar{v}-V)^2\right)\mathrm{d}z$$

The so-called dispersive terms, $\frac{1}{\rho}\frac{\partial}{\partial x}\int_{-h_x}^{\zeta}\left(\rho(\bar{u}-U)^2\right)\mathrm{d}z$, $\frac{1}{\rho}\frac{\partial}{\partial y}\int_{-h_x}^{\zeta}\rho(\bar{u}-U)(\bar{v}-V)\mathrm{d}z$, $\frac{1}{\rho}\frac{\partial}{\partial x}\int_{-h_x}^{\zeta}\rho(\bar{u}-U)(\bar{v}-V)\mathrm{d}z$ and $\frac{1}{\rho}\frac{\partial}{\partial y}\int_{-h_x}^{\zeta}\left(\rho(\bar{v}-V)^2\right)\mathrm{d}z$ represent horizontal momentum exchanges due to the depth-averaging process. Their effect is often expressed by means of a momentum correction factors, β_1, β_2 and β_3 which is calculated for a given boundary layer profile for $\bar{u}$ and $\bar{v}$, and then subsumed into the corresponding advective acceleration term to give

$$\frac{\partial(Uh)}{\partial t}+\frac{\partial(\beta_1U^2h)}{\partial x}+\frac{\partial(\beta_2UVh)}{\partial y}=fVh-\frac{1}{\rho}\int_{-h_s}^{\zeta}\frac{\partial \bar{p}}{\partial x}\mathrm{d}z+\frac{\tau_{wx}-\tau_{bx}}{\rho}$$
$$+\frac{1}{\rho}\frac{\partial}{\partial x}\int_{-h_x}^{\zeta}\left(2\mu\frac{\partial\bar{u}}{\partial x}-\rho\overline{u'u'}\right)\mathrm{d}z+\frac{1}{\rho}\frac{\partial}{\partial y}\int_{-h_x}^{\zeta}\left(\mu\left(\frac{\partial\bar{u}}{\partial y}+\frac{\partial\bar{v}}{\partial x}\right)-\rho\overline{u'v'}\right)\mathrm{d}z$$

and

$$\frac{\partial(Vh)}{\partial t}+\frac{\partial(\beta_2 UVh)}{\partial x}+\frac{\partial(\beta_3 V^2 h)}{\partial y}=-fUh-\frac{1}{\rho}\int_{-h_s}^{\zeta}\frac{\partial \bar{p}}{\partial y}\mathrm{dz}+\frac{\tau_{wy}-\tau_{by}}{\rho}$$
$$+\frac{1}{\rho}\frac{\partial}{\partial x}\int_{-h_x}^{\zeta}\left(\mu\left(\frac{\partial \bar{u}}{\partial y}+\frac{\partial \bar{v}}{\partial x}\right)-\rho\overline{u'v'}\right)\mathrm{dz}+\frac{1}{\rho}\frac{\partial}{\partial y}\int_{-h_x}^{\zeta}\left(2\mu\frac{\partial \bar{v}}{\partial y}-\rho\overline{v'v'}\right)\mathrm{dz} \tag{3.22}$$

Next, we define the effective stresses as

$$T_{xx}=\frac{1}{h}\int_{-h_x}^{\zeta}\left(2\mu\frac{\partial \bar{u}}{\partial x}-\rho\overline{u'u'}\right)\mathrm{dz},$$

$$T_{xy}=\frac{1}{h}\int_{-h_x}^{\zeta}\left(\mu\left(\frac{\partial \bar{u}}{\partial y}+\frac{\partial \bar{v}}{\partial x}\right)-\rho\overline{u'v'}\right)\mathrm{dz}, \tag{3.23}$$

and

$$T_{yy}=\frac{1}{h}\int_{-h_x}^{\zeta}\left(2\mu\frac{\partial \bar{v}}{\partial y}-\rho\overline{v'v'}\right)\mathrm{dz}\quad .$$

The effective stresses presented here have two terms: the first relates to viscous stresses, the second to the turbulent Reynolds stresses. Substituting for the effective stresses, we have

$$\frac{\partial(Uh)}{\partial t}+\frac{\partial(\beta_1 U^2 h)}{\partial x}+\frac{\partial(\beta_2 UVh)}{\partial y}=fVh-\frac{1}{\rho}\int_{-h_s}^{\zeta}\frac{\partial \bar{p}}{\partial x}\mathrm{dz}+\frac{\tau_{wx}-\tau_{bx}}{\rho}$$
$$+\frac{1}{\rho}\frac{\partial(hT_{xx})}{\partial x}+\frac{1}{\rho}\frac{\partial(hT_{xy})}{\partial y}$$

and

$$\frac{\partial(Vh)}{\partial t}+\frac{\partial(\beta_2 UVh)}{\partial x}+\frac{\partial(\beta_3 V^2 h)}{\partial y}=-fUh-\frac{1}{\rho}\int_{-h_s}^{\zeta}\frac{\partial \bar{p}}{\partial y}\mathrm{dz}+\frac{\tau_{wy}-\tau_{by}}{\rho}$$
$$+\frac{1}{\rho}\frac{\partial(hT_{xy})}{\partial x}+\frac{1}{\rho}\frac{\partial(hT_{yy})}{\partial y} \tag{3.24}$$

Replacing the Reynolds stresses by the Boussinesq approximation, the effective stresses become (noting that $\varepsilon >> \nu$, and making an approximation for the depth-averaged eddy viscosity)

$$T_{xx} = \frac{1}{h}\int_{-h_x}^{\zeta}\left(2\rho(\varepsilon+\nu)\frac{\partial \bar{u}}{\partial x}\right)\mathrm{d}z = 2\rho\bar{\varepsilon}\frac{\partial U}{\partial x}$$

$$T_{xy} = \frac{1}{h}\int_{-h_x}^{\zeta}\rho(\varepsilon+\nu)\left(\frac{\partial \bar{u}}{\partial y}+\frac{\partial \bar{v}}{\partial x}\right)\mathrm{d}z = \rho\bar{\varepsilon}\left(\frac{\partial U}{\partial y}+\frac{\partial V}{\partial x}\right) \qquad (3.25)$$

and

$$T_{yy} = \frac{1}{h}\int_{-h_x}^{\zeta}\left(2\rho(\varepsilon+\nu)\frac{\partial \bar{v}}{\partial y}\right)\mathrm{d}z = 2\rho\bar{\varepsilon}\frac{\partial V}{\partial y}\,.$$

where $\bar{\varepsilon}$ is the depth-averaged eddy viscosity. Hence, we can also write the momentum equations as

$$\frac{\partial(Uh)}{\partial t}+\frac{\partial(\beta_1 U^2 h)}{\partial x}+\frac{\partial(\beta_2 UVh)}{\partial y} = fVh-\frac{1}{\rho}\int_{-h_s}^{\zeta}\frac{\partial \bar{p}}{\partial x}\mathrm{d}z+\frac{\tau_{wx}-\tau_{bx}}{\rho} + 2\frac{\partial}{\partial x}\left(\bar{\varepsilon}h\frac{\partial U}{\partial x}\right)+\frac{\partial}{\partial y}\left(\bar{\varepsilon}h\left[\frac{\partial U}{\partial y}+\frac{\partial V}{\partial x}\right]\right)$$

and

$$\frac{\partial(Vh)}{\partial t}+\frac{\partial(\beta_2 UVh)}{\partial x}+\frac{\partial(\beta_3 V^2 h)}{\partial y} = -fUh-\frac{1}{\rho}\int_{-h_s}^{\zeta}\frac{\partial \bar{p}}{\partial y}\mathrm{d}z+\frac{\tau_{wy}-\tau_{by}}{\rho} + \frac{\partial}{\partial x}\left(\bar{\varepsilon}h\left[\frac{\partial U}{\partial y}+\frac{\partial V}{\partial x}\right]\right)+2\frac{\partial}{\partial y}\left(\bar{\varepsilon}h\frac{\partial V}{\partial y}\right) \qquad (3.26)$$

The z-direction Reynolds-averaged momentum equation simplifies to give $\frac{\partial \bar{p}}{\partial z} = -\rho g$ Integrating, and applying the dynamic pressure boundary condition that the pressure at the free surface is atmospheric (i.e. $\bar{p} = p_a$ at $z = \zeta$, leads to the **hydrostatic** pressure distribution given by $\bar{p} = \rho g(\zeta - z) + p_a$ where z is taken upwards from still water level. This expression is differentiated with respect to *x* and *y*, giving

$$\frac{\partial \bar{p}}{\partial x} = \rho g \frac{\partial \zeta}{\partial x} + \frac{\partial p_a}{\partial x} \qquad \text{and} \qquad \frac{\partial \bar{p}}{\partial y} = \rho g \frac{\partial \zeta}{\partial y} + \frac{\partial p_a}{\partial y}. \qquad (3.27)$$

so that the pressure gradient terms in the Reynolds-averaged Navier-Stokes equations may be replaced by free surface elevation gradients, such that

$$\frac{\partial (Uh)}{\partial t} + \frac{\partial (\beta_1 U^2 h)}{\partial x} + \frac{\partial (\beta_2 UVh)}{\partial y} = fVh - gh\frac{\partial \zeta}{\partial x} - \frac{h}{\rho}\frac{\partial p_a}{\partial x} + \frac{\tau_{wx} - \tau_{bx}}{\rho} + 2\frac{\partial}{\partial x}\left(\bar{\varepsilon} h \frac{\partial U}{\partial x}\right) + \frac{\partial}{\partial y}\left(\bar{\varepsilon} h \left[\frac{\partial U}{\partial y} + \frac{\partial V}{\partial x}\right]\right)$$

and

$$\frac{\partial (Vh)}{\partial t} + \frac{\partial (\beta_2 UVh)}{\partial x} + \frac{\partial (\beta_3 V^2 h)}{\partial y} = -fUh - gh\frac{\partial \zeta}{\partial y} - \frac{h}{\rho}\frac{\partial p_a}{\partial y} + \frac{\tau_{wy} - \tau_{by}}{\rho} + \frac{\partial}{\partial x}\left(\bar{\varepsilon} h \left[\frac{\partial U}{\partial y} + \frac{\partial V}{\partial x}\right]\right) + 2\frac{\partial}{\partial y}\left(\bar{\varepsilon} h \frac{\partial V}{\partial y}\right) \qquad (3.28)$$

In small-ish domains, we can neglect the Coriolis terms. Also, assuming a negligible atmospheric pressure gradient, we obtain

$$\frac{\partial (Uh)}{\partial t} + \frac{\partial (\beta_1 U^2 h)}{\partial x} + \frac{\partial (\beta_2 UVh)}{\partial y} = -gh\frac{\partial \zeta}{\partial x} + \frac{\tau_{wx} - \tau_{bx}}{\rho} + 2\frac{\partial}{\partial x}\left(\bar{\varepsilon} h \frac{\partial U}{\partial x}\right) + \frac{\partial}{\partial y}\left(\bar{\varepsilon} h \left[\frac{\partial U}{\partial y} + \frac{\partial V}{\partial x}\right]\right)$$

and

$$\frac{\partial (Vh)}{\partial t} + \frac{\partial (\beta_2 UVh)}{\partial x} + \frac{\partial (\beta_3 V^2 h)}{\partial y} = -gh\frac{\partial \zeta}{\partial y} + \frac{\tau_{wy} - \tau_{by}}{\rho} + \frac{\partial}{\partial x}\left(\bar{\varepsilon} h \left[\frac{\partial U}{\partial y} + \frac{\partial V}{\partial x}\right]\right) + 2\frac{\partial}{\partial y}\left(\bar{\varepsilon} h \frac{\partial V}{\partial y}\right) \qquad (3.29)$$

Simplifying the turbulent diffusion terms further, we may write

$$\frac{\partial(Uh)}{\partial t}+\frac{\partial(\beta_1 U^2 h)}{\partial x}+\frac{\partial(\beta_2 UVh)}{\partial y}=-gh\frac{\partial\zeta}{\partial x}+\frac{\tau_{wx}-\tau_{bx}}{\rho}+\bar{\varepsilon}h\left[\frac{\partial^2 U}{\partial x^2}+\frac{\partial^2 V}{\partial y^2}\right]$$

and

$$\frac{\partial(Vh)}{\partial t}+\frac{\partial(\beta_2 UVh)}{\partial x}+\frac{\partial(\beta_3 V^2 h)}{\partial y}=-gh\frac{\partial\zeta}{\partial y}+\frac{\tau_{wy}-\tau_{by}}{\rho}+\bar{\varepsilon}h\left[\frac{\partial^2 U}{\partial x^2}+\frac{\partial^2 V}{\partial y^2}\right] \quad (3.30)$$

For laminar flow, $\beta_1=\beta_2=\beta_3=1.2$. For turbulent flow where the velocity profile follows a 1/7th power law, $\beta_1=\beta_2=\beta_3=1.016$, and is often assumed to be unity (see e.g. white, 2003). The surface wind stress components and the bed stress components are estimated from empirical formulae. The wind stress components are calculated from $\tau_{wx}=\rho_a c_w W_x\sqrt{W_x^2+W_y^2}$ and $\tau_{wy}=\rho_a c_w W_y\sqrt{W_x^2+W_y^2}$ in which ρ_a is the density of air, c_w is the air-water resistance coefficient and W_x and W_y are the wind velocity components. A typical value of c_w is 0.0026. The bed stresses are given by $\tau_{bx}=\rho c_f U\sqrt{U^2+V^2}$ and $\tau_{by}=\rho c_f V\sqrt{U^2+V^2}$, in which the bed friction coefficient, $c_f=\frac{g}{C^2}=\frac{gn^2}{h^{1/3}}$ where C is the Chézy coefficient (typically in the range from 30 to 100 $m^{1/2}s^{-1}$) and n is the Manning coefficient (typically varying from 0.015 to 0.04) (see e.g. Ven Te Chow, 1959). In practice, the depth-mean eddy viscosity $\bar{\varepsilon}$ can be either set empirically or by use of a turbulence model (see e.g. Rodi, 1984). Assuming a linear shear stress distribution and logarithmic velocity distribution with depth, the depth-mean eddy viscosity may be shown to be

$$\bar{\varepsilon}=\frac{\kappa\sqrt{g}\sqrt{U^2+V^2}}{6C}h$$

in which κ is the von Kármán constant.

3.4. Direct derivation of the Shallow Water Equations

Shallow water equations can be obtained based on force/momentum balance method. Figure 3.3 shows the element extending through the depth, across which we will

consider mass and impulse-momentum balances. The elemental volume has plan dimensions Δx and Δy. The total depth is $h = h_s + \zeta$.

In Figure 3.3, the distance between the bed and the horizontal datum is z_b, u and v are the flow velocity components which are assumed uniform over the depth, g is the acceleration due to gravity, and ρ is the density of water. The derivation follows that of Abbott (1979):

Mass of water in the element is $\rho h \Delta x \Delta y$.

Accumulation of water mass in the element in time Δt is $\frac{\partial \rho h}{\partial t} \Delta x \Delta y \Delta t$.

Mass flux into the element in time Δt is $\rho u h \Delta y \Delta t + \rho v h \Delta x \Delta t$.

Mass flux out of the element in time Δt is $(\rho u h + \frac{\partial \rho u h}{\partial x} \Delta x)\ \Delta y \Delta t$ + $(\rho v h + \frac{\partial \rho v h}{\partial y} \Delta y)\ \Delta x \Delta t$.

From the mass balance, we have:

$$\frac{\partial \rho h}{\partial t} \Delta x \Delta y \Delta t = \rho u h \Delta y \Delta t - (\rho u h + \frac{\partial (\rho u h)}{\partial x} \Delta x) \Delta y \Delta t + \rho v h \Delta x \Delta t - (\rho v h + \frac{\partial (\rho v h)}{\partial y} \Delta y) \Delta x \Delta t \quad .$$

(3.31)

Rewriting this, the mass conservation equation is

$$\frac{\partial \rho h}{\partial t} + \frac{\partial (\rho u h)}{\partial x} + \frac{\partial (\rho v h)}{\partial y} = 0 . \qquad (3.32)$$

Assuming constant density for an incompressible fluid, we obtain the continuity equation

$$\frac{\partial h}{\partial t} + \frac{\partial (u h)}{\partial x} + \frac{\partial (v h)}{\partial y} = 0 . \qquad (3.33)$$

Denoting uh and vh by q_x and q_y, the continuity equation becomes:

$$\frac{\partial h}{\partial t} + \frac{\partial q_x}{\partial x} + \frac{\partial q_y}{\partial y} = 0 . \qquad (3.34)$$

Next, consider the momentum balance in the *x*-direction by applying Figure 3.4.

Pressure thrust on the left hand side of the element (inlet): $\frac{1}{2}\rho g h^2 \Delta y \Delta t$.

Pressure thrust on right hand side of the element (outlet) is: $-(\frac{1}{2}\rho g h^2 + \frac{\partial(\frac{1}{2}\rho g h^2)}{\partial x}\Delta x)\Delta y \Delta t$.

Bed friction is: $-\tau_{bx}\Delta x \Delta y \Delta t$ where τ_{bx} is the bed stress component in the *x*-direction.

Wind resistance is: $\tau_{wx}\Delta x \Delta y \Delta t$ where τ_{wx} is the wind stress component in the *x*-direction.

Weight component is: $-\rho g h \frac{\partial z_b}{\partial x}\Delta x \Delta y \Delta t$.

Coriolis component: $f\rho h \Delta x \Delta y v \Delta t$ in which the Coriolis coefficient $f = 2\omega \sin\phi$ where ω is angular speed of Earth's rotation (~7.3 x 10^{-5} rad/s) and ϕ is the latitude

Accumulation of momentum in element in time Δt is given by: $\frac{\partial(\rho u h)}{\partial t}\Delta x \Delta y \Delta t$.

Momentum flux entering west of the element (inlet) is: $\rho u^2 h \Delta y \Delta t$.

Momentum flux leaving east of the element (outlet) is: $\left(\rho u^2 h + \frac{\partial(\rho u^2 h)}{\partial x}\Delta x\right)\Delta y \Delta t$.

Momentum flux entering south of the element is: $\rho u v h \Delta x \Delta t$.

Momentum flux leaving at north of the element is: $(\rho u v h + \frac{\partial(\rho u v h)}{\partial y}\Delta y)\Delta x \Delta t$.

By equating the net force components to the difference between the momentum flux entering to that entering, we obtain the shallow water momentum equation in the *x*-direction as:

$$\frac{1}{2}\rho g h^2 \Delta y \Delta t - (\frac{1}{2}\rho g h^2 + \frac{\partial(\frac{1}{2}\rho g h^2)}{\partial x}\Delta x)\Delta y \Delta t + \tau_{wx}\Delta x \Delta y \Delta t - \tau_{bx}\Delta x \Delta y \Delta t - \rho g h \frac{\partial z_b}{\partial x}\Delta x \Delta y \Delta t + f\rho h \Delta x \Delta y v \Delta t =$$

$$= \frac{\partial(\rho u h)}{\partial t}\Delta x \Delta y \Delta t + (\rho u^2 h + \frac{\partial(\rho u^2 h)}{\partial x}\Delta x)\Delta y \Delta t - \rho u^2 h \Delta y \Delta t + (\rho u v h + \frac{\partial(\rho u v h)}{\partial y}\Delta y)\Delta x \Delta t - \rho u v h \Delta x \Delta t.$$

Rearranging the above equation leads to:

$$-\frac{\partial(\frac{1}{2}\rho g h^2)}{\partial x}\Delta x\Delta y\Delta t+\tau_{wx}\Delta x\Delta y\Delta t-\tau_{bx}\Delta x\Delta y\Delta t-\rho g h\frac{\partial z_b}{\partial x}\Delta x\Delta y\Delta t+ f\rho h\Delta x\Delta y v\Delta t=$$

$$=\frac{\partial(\rho u h)}{\partial t}\Delta x\Delta y\Delta t+\frac{\partial(\rho u^2 h)}{\partial x}\Delta x\Delta y\Delta t+\frac{\partial(\rho u v h)}{\partial y}\Delta y\Delta x\Delta t\ . \qquad (3.35)$$

or

$$\frac{\partial(\rho u h)}{\partial t}+\frac{\partial(\rho u^2 h+\frac{1}{2}\rho g h^2)}{\partial x}+\frac{\partial(\rho u v h)}{\partial y}=-\rho g h\frac{\partial z_b}{\partial x}+\tau_{wx}-\tau_{bx}+ f\rho h v\ . \qquad (3.36)$$

For an incompressible flow,

$$\frac{\partial(u h)}{\partial t}+\frac{\partial(u^2 h+\frac{1}{2} g h^2)}{\partial x}+\frac{\partial(u v h)}{\partial y}=-g h\frac{\partial z_b}{\partial x}+\frac{\tau_{wx}-\tau_{bx}}{\rho}+ f h v\ . \qquad (3.37)$$

Substituting discharge (q_x and q_y) for the velocity components, the momentum equation in the *x* direction becomes:

$$\frac{\partial q_x}{\partial t}+\frac{1}{h}\frac{\partial(q_x{}^2+\frac{1}{2} g h^2)}{\partial x}+\frac{1}{h}\frac{\partial(q_x q_y)}{\partial y}=-g h\frac{\partial z_b}{\partial x}+\frac{\tau_{wx}-\tau_{bx}}{\rho}+ f h v\ . \qquad (3.38)$$

A similar procedure is applied to derive the momentum equation in the *y*-direction, in the following forms

$$\frac{\partial(\rho v h)}{\partial t}+\frac{\partial(\rho v^2 h+\frac{1}{2}\rho g h^2)}{\partial y}+\frac{\partial(\rho u v h)}{\partial x}=-\rho g h\frac{\partial z_b}{\partial y}+\tau_{wy}-\tau_{by}- f\rho h u\ . \qquad (3.39)$$

For incompressible flow,

$$\frac{\partial(v h)}{\partial t}+\frac{\partial(v^2 h+\frac{1}{2} g h^2)}{\partial y}+\frac{\partial(u v h)}{\partial x}=-g h\frac{\partial z_b}{\partial y}+\frac{\tau_{wy}-\tau_{by}}{\rho}- f h u\ . \qquad (3.40)$$

or

$$\frac{\partial q_y}{\partial t}+\frac{1}{h}\frac{\partial(q_y{}^2+\frac{1}{2}gh^2)}{\partial y}+\frac{1}{h}\frac{\partial(q_x q_y)}{\partial x}=-gh\frac{\partial z_b}{\partial y}+\frac{\tau_{wy}-\tau_{by}}{\rho}-fhu\,. \qquad (3.41)$$

In summary, the shallow water equations have been determined by two methods. The first was based on depth-integration of the fundamental continuity and Navier-Stokes equations after Reynolds-averaging. The second method applies mass and force/momentum balances assuming uniform velocity profile over the depth to determine directly the shallow water equations, but neglected the momentum correction factor and effective stress terms. The shallow water equations are summarized below:

$$\frac{\partial\zeta}{\partial t}+\frac{\partial(Uh)}{\partial x}+\frac{\partial(Vh)}{\partial y}=0\,. \qquad (3.42)$$

$$\frac{\partial(Uh)}{\partial t}+\frac{\partial(\beta_1 U^2 h)}{\partial x}+\frac{\partial(\beta_2 UVh)}{\partial y}=fVh-gh\frac{\partial\zeta}{\partial x}+\frac{\tau_{wx}-\tau_{bx}}{\rho}+\frac{1}{\rho}\frac{\partial(hT_{xx})}{\partial x}+\frac{1}{\rho}\frac{\partial(hT_{xy})}{\partial y} \qquad (3.43)$$

$$\frac{\partial(Vh)}{\partial t}+\frac{\partial(\beta_2 UVh)}{\partial x}+\frac{\partial(\beta_3 V^2 h)}{\partial y}=-fUh-gh\frac{\partial\zeta}{\partial y}+\frac{\tau_{wy}-\tau_{by}}{\rho}+\frac{1}{\rho}\frac{\partial(hT_{xy})}{\partial x}+\frac{1}{\rho}\frac{\partial(hT_{yy})}{\partial y} \qquad (3.44)$$

where

$$T_{xx}=\frac{1}{h}\int_{-h_x}^{\zeta}\left(2\rho(\varepsilon+\nu)\frac{\partial\bar{u}}{\partial x}\right)\mathrm{d}z=2\rho\bar{\varepsilon}\frac{\partial U}{\partial x}\ ,$$

$$T_{xy}=\frac{1}{h}\int_{-h_x}^{\zeta}\rho(\varepsilon+\nu)\left(\frac{\partial\bar{u}}{\partial y}+\frac{\partial\bar{v}}{\partial x}\right)\mathrm{d}z=\rho\bar{\varepsilon}\left(\frac{\partial U}{\partial y}+\frac{\partial V}{\partial x}\right)\ ,$$

$$T_{yy}=\frac{1}{h}\int_{-h_x}^{\zeta}\left(2\rho(\varepsilon+\nu)\frac{\partial\bar{v}}{\partial y}\right)\mathrm{d}z=2\rho\bar{\varepsilon}\frac{\partial V}{\partial y}\ .$$

3.5 Transformed Governing Shallow Water Equations:

In order to solve the shallow water equations on a curvilinear grid, the equations must themselves be transformed into the mapped system, using the chain rule to preserve area by means of the transformation Jacobian. Here, we define the

transformed coordinate directions as ξ and η, following Thomson *et al*. (1977). The basic equations for transforming *x* and *y* to ξ and η are:

$$f_x = \frac{\partial f}{\partial x} = \frac{\partial(f,y)}{\partial(\xi,\eta)} \div \frac{\partial(x,y)}{\partial(\xi,\eta)} = \frac{1}{J}(y_\eta f_\xi - f_\eta y_\xi)\,. \tag{3.45}$$

$$f_y = \frac{\partial f}{\partial y} = \frac{\partial(x,f)}{\partial(\xi,\eta)} \div \frac{\partial(x,y)}{\partial(\xi,\eta)} = \frac{1}{J}(x_\xi f_\eta - f_\zeta x_\eta)\,. \tag{3.46}$$

where f is a differentiable function of *x* and *y*, J is the Jacobian ($J = x_\xi y_\eta - x_\eta y_\xi$), and subscript notation is used for differentiation. Utilising equations (3.45)-(3.46) the transformed shallow water equations are obtained (see e.g. Barber 1990) as:

$$\frac{\partial\zeta}{\partial t} + \frac{1}{J}\left(y_\eta \frac{\partial(Uh)}{\partial\xi} - y_\xi \frac{\partial(Uh)}{\partial\eta} + x_\xi \frac{\partial(Vh)}{\partial\eta} - x_\eta \frac{\partial(Vh)}{\partial\xi} \right) = 0\,. \tag{3.47}$$

$$\frac{\partial(Uh)}{\partial t} + \frac{1}{J}\left(y_\eta \frac{\partial(\beta U^2 h)}{\partial\xi} - y_\xi \frac{\partial(\beta U^2 h)}{\partial\eta} + x_\xi \frac{\partial(\beta UVh)}{\partial\eta} - x_\eta \frac{\partial(\beta UVh)}{\partial\xi} \right) - fVh + \frac{gh}{J}(y_\eta \frac{\partial\zeta}{\partial\xi} - y_\xi \frac{\partial\zeta}{\partial\eta}) -$$

$$- \frac{\tau_{wx} - \tau_{bx}}{\rho} - \frac{1}{\rho J}(y_\eta \frac{\partial(hT_{xx})}{\partial\xi} - y_\xi \frac{\partial(hT_{xx})}{\partial\eta} + x_\xi \frac{\partial(hT_{xy})}{\partial\eta} - x_\eta \frac{\partial(hT_{xy})}{\partial\xi} \Bigg) = 0\,. \tag{3.48}$$

and

$$\frac{\partial(Vh)}{\partial t} + \frac{1}{J}\left(y_\eta \frac{\partial(\beta UVh)}{\partial\xi} - y_\xi \frac{\partial(\beta UVh)}{\partial\eta} + x_\xi \frac{\partial(\beta V^2 h)}{\partial\eta} - x_\eta \frac{\partial(\beta V^2 h)}{\partial\xi} \right) + fUh + \frac{gh}{J}(x_\xi \frac{\partial\zeta}{\partial\eta} - x_\eta \frac{\partial\zeta}{\partial\xi}) -$$

$$- \frac{\tau_{wy} - \tau_{by}}{\rho} - \frac{1}{\rho J}(y_\eta \frac{\partial(hT_{xy})}{\partial\xi} - y_\xi \frac{\partial(hT_{xy})}{\partial\eta} + x_\xi \frac{\partial(hT_{yy})}{\partial\eta} - x_\eta \frac{\partial(hT_{yy})}{\partial\xi} \Bigg) = 0\,. \tag{3.49}$$

Where

$$T_{xx} = \frac{2\rho\bar{\varepsilon}}{J}(y_\eta \frac{\partial U}{\partial\xi} - y_\xi \frac{\partial U}{\partial\eta})\ ,$$

$$T_{xy} = \frac{\rho\bar{\varepsilon}}{J}\left(x_\xi \frac{\partial U}{\partial\eta} - x_\eta \frac{\partial U}{\partial\xi} + y_\eta \frac{\partial V}{\partial\xi} - y_\xi \frac{\partial V}{\partial\eta} \right),$$

$$T_{yy} = \frac{2\rho\bar{\varepsilon}}{J}(x_\xi \frac{\partial V}{\partial\eta} - x_\eta \frac{\partial V}{\partial\xi})\ .$$

3.6 Boundary Conditions

As initial-boundary-value partial differential equations, the hydrodynamics equations require appropriate boundary conditions for the dependent variables (depth and velocity components) at walls and open boundaries in order to determine flow features as they evolve in open channels. Herein, both slip and no-slip conditions are applied at solid walls. The solid wall slip condition is:

- Velocity component normal to the wall, $u_n = 0$
- Gradient of velocity component tangential to the wall, $\frac{\partial u_t}{\partial n} = 0$
- Depth given by extrapolation

The solid wall no-slip condition is:

- Velocity component normal to the wall, $u_n = 0$
- Velocity component tangential to the wall, $u_t = 0$
- Depth given by extrapolation

Also by considering subcritical flow

The open (transmissive) condition at inflow boundary is:

- Prescribed inflow normal velocity component, $u_n =$ Prescribed
- Zero tangential velocity component, $u_t = 0$
- Zero gradient in free surface elevation $\frac{\partial h}{\partial n} = 0$.

For subcritical flow, The open (transmissive) condition at outflow boundary is:

- Zero gradient in outflow normal velocity component, $\frac{\partial u_n}{\partial n} = 0$
- Zero gradient in outflow tangential velocity component, $\frac{\partial u_t}{\partial n} = 0$
- Prescribed depth

3.7 Numerical implementation

The Cartesian shallow water equations (3.42), (3.43) and (3.44) and the curvilinear shallow water equations (3.47), (3.48) and (3.49) are first discretized in space using second-order central differences. An example of the computational grid layout is depicted in Figure 3.5. The numerical solution is then obtained using an explicit time marching scheme with the integrations performed using either second-order Adams-Bashforth or fourth-order Runge-Kutta methods.

3.7.1 Finite Difference Solution of Cartesian Shallow Water Equations

In this case, the grid is rectangular, such that $x = i\Delta x$ and $y = j\Delta y$, where i=1, 2, … $i_{\max}$ and j = 1, 2, … $j_{\max}$ are indices and Δx and Δy are the grid intervals in the x- and y-directions. Time is set such that $t = k\Delta t$. Following Fletcher (1997), Abbott and Basco (1989) and others, the Cartesian shallow equations (3.42), (3.43) and (3.44) are discretized using central differences, giving

$$\left.\frac{\partial(\zeta)}{\partial t}\right|_{i,j}^{k} = -\left(\left.\frac{\partial(Uh)}{\partial x}\right|_{i,j}^{k} + \left.\frac{\partial(Vh)}{\partial y}\right|_{i,j}^{k} \right). \qquad (3.50)$$

$$\left.\frac{\partial(\zeta)}{\partial t}\right|_{i,j}^{k} = -\left(\frac{U_{i+1,j}h_{i+1,j} - U_{i-1,j}h_{i-1,j}}{2\Delta x} + \frac{V_{i,j+1}h_{i,j+1} - V_{i,j-1}h_{i,j-1}}{2\Delta y} \right). \qquad (3.51)$$

$$\left.\frac{\partial(Uh)}{\partial t}\right|_{i,j}^{k} = -\left.\frac{\partial(\beta U^2 h)}{\partial x}\right|_{i,j}^{k} - \left.\frac{\partial(\beta UVh)}{\partial y}\right|_{i,j}^{k} + \left(fVh\right)_{i,j}^{k} - \left(gh\left.\frac{\partial\zeta}{\partial x}\right|_{i,j}^{k} + \frac{\tau_{wx} - \tau_{bx}}{\rho} + \left(\frac{1}{\rho}\left.\frac{\partial(hT_{xx})}{\partial x}\right|_{i,j}^{k} + \left(\frac{1}{\rho}\left.\frac{\partial(hT_{xy})}{\partial y}\right|_{i,j}^{k} \right.\right.\right.\right.$$

$$\left.\frac{\partial(Uh)}{\partial t}\right|_{i,j}^{k} = -\left(\beta\frac{U^2{}_{i+1,j}h_{i+1,j} - U^2{}_{i-1,j}h_{i-1,j}}{2\Delta x} \right) + \left(\beta\frac{U_{i,j+1}V_{i,j+1}h_{i,j+1} - U_{i,j-1}V_{i,j-1}h_{i,j-1}}{2\Delta y} \right) + fV_{i,j}h_{i,j} - gh_{i,j}\frac{\zeta_{i+1,j} - \zeta_{i-1,j}}{2\Delta x} +$$

$$+ \frac{\tau_{wx} - \tau_{bx}}{\rho} + \frac{1}{\rho}\frac{h_{i+1,j} - h_{i-1,j}}{2\Delta x} 2\rho\bar{\varepsilon}\frac{U_{i+1,j} - 2U_{i,j} - U_{i-1,j}}{\Delta x^2} + \frac{1}{\rho}\frac{h_{i,j+1} - h_{i,j-1}}{2\Delta y}\rho\bar{\varepsilon}\frac{U_{i,j+1} - 2U_{i,j} - U_{i,j-1}}{\Delta y^2} +$$

$$+ \frac{1}{\rho}\frac{h_{i,j+1} - h_{i,j-1}}{2\Delta y}\rho\bar{\varepsilon}\frac{1}{4\Delta x\Delta y}\left((V_{i+1,j+1} - V_{i+1,j-1}) - (V_{i-1,j+1} - V_{i-1,j-1}) \right). \qquad (3.52)$$

and

$$\left.\frac{\partial(Vh)}{\partial t}\right|_{i,j}^{k} = -\left.\frac{\partial(\beta UVh)}{\partial x}\right|_{i,j}^{k} - \left.\frac{\partial(\beta V^2 h)}{\partial y}\right|_{i,j}^{k} - \left(fUh\right)_{i,j}^{k} - \left.\left(gh\frac{\partial\zeta}{\partial y}\right)\right|_{i,j}^{k} + \frac{\tau_{wy}-\tau_{by}}{\rho} + \left.\left(\frac{1}{\rho}\frac{\partial(hT_{xy})}{\partial x}\right)\right|_{i,j}^{k} + \left.\left(\frac{1}{\rho}\frac{\partial(hT_{yy})}{\partial y}\right)\right|_{i,j}^{k}$$

$$\left.\frac{\partial(Vh)}{\partial t}\right|_{i,j}^{k} = -\left(\beta\frac{U_{i+1,j}V_{i+1,j}h_{i+1,j} - U_{i-1,j}V_{i-1,j}h_{i-1,j}}{2\Delta x}\right) - \left(\beta\frac{V^2{}_{i,j+1}h_{i,j+1} - V^2{}_{i,j-1}h_{i,j-1}}{2\Delta y}\right) - fU_{i,j}h_{i,j} - gh_{i,j}\frac{\zeta_{i,j+1}-\zeta_{i,j-1}}{2\Delta y} +$$

$$+\frac{\tau_{wy}-\tau_{by}}{\rho} + \frac{1}{\rho}\frac{h_{i+1,j}-h_{i-1,j}}{2\Delta x}\rho\bar{\varepsilon}\left(\frac{1}{4\Delta x\Delta y}\left((U_{i+1,j+1}-U_{i+1,j-1})-(U_{i-1,j+1}-U_{i-1,j-1})\right)+\frac{V_{i+1,j}-2V_{i,j}-V_{i-1,j}}{\Delta x^2}\right)+$$

$$+\frac{1}{\rho}\frac{h_{i,j+1}-h_{i,j-1}}{2\Delta y}2\rho\bar{\varepsilon}\frac{V_{i,j+1}-2V_{i,j}-V_{i,j-1}}{\Delta y^2}. \qquad (3.53)$$

3.7.2 Finite Difference Solution of Transformed Shallow Water Equations

The transformed shallow water equations are solved on a curvilinear grid, whose lines are defined such that $\xi = i\Delta\xi$ and $\eta = j\Delta\eta$, where i=1, 2, … $i_{\max}$ and j = 1, 2, … $j_{\max}$ are indices and $\Delta\xi$ and $\Delta\eta$ are the grid intervals in the ξ- and η-directions. Time is set such that $t = k\Delta t$. Following Thompson *et al.*, 1974, the transformed shallow water equations (3.47), (3.48) and (3.49) are discretized using central differences.

The discretized version of the transformed continuity equation,

$$\left.\frac{\partial\zeta}{\partial t}\right|_{i,j}^{k} = -\frac{1}{J}\left(y_\eta\Big|_{i,j}^{k}\left.\frac{\partial(Uh)}{\partial\xi}\right|_{i,j}^{k} - y_\xi\Big|_{i,j}^{k}\left.\frac{\partial(Uh)}{\partial\eta}\right|_{i,j}^{k} + x_\xi\Big|_{i,j}^{k}\left.\frac{\partial(Vh)}{\partial\eta}\right|_{i,j}^{k} - x_\eta\Big|_{i,j}^{k}\left.\frac{\partial(Vh)}{\partial\xi}\right|_{i,j}^{k}\right),$$

becomes

$$\left.\frac{\partial(\zeta)}{\partial t}\right|_{i,j}^{k} = -\frac{1}{J}\left(\frac{y_{i,j+1}-y_{i,j-1}}{2\Delta\eta}\frac{U_{i+1,j}h_{i+1,j}-U_{i-1,j}h_{i-1,j}}{2\Delta\xi} - \frac{y_{i+1,j}-y_{i-1,j}}{2\Delta\xi}\frac{U_{i,j+1}h_{i,j+1}-U_{i,j-1}h_{i,j-1}}{2\Delta\eta} +\right.$$

$$\left.+\frac{x_{i+1,j}-x_{i-1,j}}{2\Delta\xi}\frac{V_{i,j+1}h_{i,j+1}-V_{i,j-1}h_{i,j-1}}{2\Delta\eta} - \frac{x_{i,j+1}-x_{i,j-1}}{2\Delta\eta}\frac{V_{i+1,j}h_{i+1,j}-V_{i-1,j}h_{i-1,j}}{2\Delta\xi}\right). \qquad (3.54)$$

Similarly, the discretized transformed momentum equation in the *x*-direction,

$$\left.\frac{\partial(Uh)}{\partial t}\right|_{i,j}^{k} = -\frac{1}{J}\left(y_\eta\Big|_{i,j}^{k}\left.\frac{\partial(\beta U^2 h)}{\partial \xi}\right|_{i,j}^{k} - y_\xi\Big|_{i,j}^{k}\left.\frac{\partial(\beta U^2 h)}{\partial \eta}\right|_{i,j}^{k} + x_\xi\Big|_{i,j}^{k}\left.\frac{\partial(\beta UVh)}{\partial \eta}\right|_{i,j}^{k} - x_\eta\Big|_{i,j}^{k}\left.\frac{\partial(\beta UVh)}{\partial \xi}\right|_{i,j}^{k}\right) + fVh\Big|_{i,j}^{k} -$$

$$-\left.\frac{gh}{J}\right|_{i,j}^{k}\left(y_\eta\Big|_{i,j}^{k}\left.\frac{\partial\zeta}{\partial\xi}\right|_{i,j}^{k} - y_\xi\Big|_{i,j}^{k}\left.\frac{\partial\zeta}{\partial\eta}\right|_{i,j}^{k}\right) + \frac{\tau_{wx}-\tau_{bx}}{\rho} + \frac{1}{\rho J}\left((y_\eta\Big|_{i,j}^{k}\left.\frac{\partial(hT_{xx})}{\partial\xi}\right|_{i,j}^{k} - y_\xi\Big|_{i,j}^{k}\left.\frac{\partial(hT_{xx})}{\partial\eta}\right|_{i,j}^{k}\right) +$$

$$+\frac{1}{\rho J}\left(x_\xi\Big|_{i,j}^{k}\left.\frac{\partial(hT_{xy})}{\partial\eta}\right|_{i,j}^{k} - x_\eta\Big|_{i,j}^{k}\left.\frac{\partial(hT_{xy})}{\partial\xi}\right|_{i,j}^{k}\right),$$

is written

$$\left.\frac{\partial(Uh)}{\partial t}\right|_{i,j}^{k} = -\frac{1}{J}\left(\frac{y_{i,j+1}-y_{i,j-1}}{2\Delta\eta}\beta\frac{U^2{}_{i+1,j}h_{i+1,j} - U^2{}_{i-1,j}h_{i-1,j}}{2\Delta\xi} - \frac{y_{i+1,j}-y_{i-1,j}}{2\Delta\xi}\beta\frac{U^2{}_{i,j+1}h_{i,j+1} - U^2{}_{i,j-1}h_{i,j-1}}{2\Delta\eta} + \right.$$

$$\left. + \frac{x_{i+1,j}-x_{i-1,j}}{2\Delta\xi}\beta\frac{U_{i,j+1}V_{i,j+1}h_{i,j+1} - U_{i,j-1}V_{i,j-1}h_{i,j-1}}{2\Delta\eta} - \frac{x_{i,j+1}-x_{i,j-1}}{2\Delta\eta}\beta\frac{U_{i+1,j}V_{i+1,j}h_{i+1,j} - U_{i-1,j}V_{i-1,j}h_{i-1,j}}{2\Delta\xi}\right) +$$

$$+ fV_{i,j}h_{i,j} - g\frac{h_{i,j}}{J}\left(\frac{y_{i,j+1}-y_{i,j-1}}{2\Delta\eta}\frac{\zeta_{i+1,j}-\zeta_{i-1,j}}{2\Delta\xi} - \frac{y_{i+1,j}-y_{i-1,j}}{2\Delta\xi}\frac{\zeta_{i,j+1}-\zeta_{i,j-1}}{2\Delta\eta}\right) + \frac{\tau_{wx}-\tau_{bx}}{\rho} +$$

$$+\frac{1}{\rho J}\left(\frac{y_{i,j+1}-y_{i,j-1}}{2\Delta\eta}\frac{h_{i+1,j}-h_{i-1,j}}{2\Delta\xi}\frac{2\rho\bar{\varepsilon}}{J}\left(\frac{1}{4\Delta\xi\Delta\eta}\left(y_{i+1,j+1}-y_{i+1,j-1}-y_{i-1,j+1}-y_{i-1,j-1}\right)\frac{U_{i+1,j}-2U_{i,j}-U_{i-1,j}}{\Delta\xi^2} - \right.\right.$$

$$\left.\left. \frac{y_{i+1,j}-2y_{i,j}-y_{i-1,j}}{\Delta\xi^2}\frac{1}{4\Delta\xi\Delta\eta}\left(U_{i+1,j+1}-U_{i+1,j-1}-U_{i-1,j+1}-U_{i-1,j-1}\right)\right)\right) + \frac{1}{\rho J}\left(-\frac{y_{i+1,j}-y_{i-1,j}}{2\Delta\xi}\frac{h_{i,j+1}-h_{i,j-1}}{2\Delta\eta}\frac{2\rho\bar{\varepsilon}}{J}\right.$$

$$\left(\frac{y_{i,j+1}-2y_{i,j}-y_{i,j-1}}{\Delta\eta^2}\frac{1}{4\Delta\xi\Delta\eta}\left(U_{i+1,j+1}-U_{i+1,j-1}-U_{i-1,j+1}-U_{i-1,j-1}\right) - \frac{1}{4\Delta\xi\Delta\eta}\left(y_{i+1,j+1}-y_{i+1,j-1}-y_{i-1,j+1}-y_{i-1,j-1}\right)\right.$$

$$\left.\left.\frac{U_{i,j+1}-2U_{i,j}-U_{i,j-1}}{\Delta\eta^2}\right) + \frac{1}{\rho J}\left(-\frac{x_{i+1,j}-x_{i-1,j}}{2\Delta\xi}\frac{h_{i,j+1}-h_{i,j-1}}{2\Delta\eta}\frac{\rho\bar{\varepsilon}}{J}\left(\frac{1}{4\Delta\xi\Delta\eta}\left(x_{i+1,j+1}-x_{i+1,j-1}-x_{i-1,j+1}-x_{i-1,j-1}\right)\right.\right.\right.$$

$$\left.\left.\left.\frac{U_{i,j+1}-2U_{i,j}-U_{i,j-1}}{\Delta\eta^2} - \frac{x_{i,j+1}-2x_{i,j}-x_{i,j-1}}{\Delta\eta^2}\frac{1}{4\Delta\xi\Delta\eta}\left(U_{i+1,j+1}-U_{i+1,j-1}-U_{i-1,j+1}-U_{i-1,j-1}\right) + \frac{y_{i,j+1}-2y_{i,j}-y_{i,j-1}}{\Delta\eta^2}\right.\right.\right.$$

$$\left.\left.\left.\frac{1}{4\Delta\xi\Delta\eta}\left(V_{i+1,j+1}-V_{i+1,j-1}-V_{i-1,j+1}-V_{i-1,j-1}\right) - \frac{1}{4\Delta\xi\Delta\eta}\left(y_{i+1,j+1}-y_{i+1,j-1}-y_{i-1,j+1}-y_{i-1,j-1}\right)\frac{V_{i,j+1}-2V_{i,j}-V_{i,j-1}}{\Delta\eta^2}\right)\right)\right)$$

$$+\frac{1}{\rho J}\left(-\frac{x_{i,j+1}-x_{i,j-1}}{2\Delta\eta}\frac{h_{i+1,j}-h_{i-1,j}}{2\Delta\xi}\frac{\rho\bar{\varepsilon}}{J}\left(\frac{x_{i+1,j}-2x_{i,j}-x_{i-1,j}}{\Delta\xi^2}\frac{1}{4\Delta\xi\Delta\eta}\left(U_{i+1,j+1}-U_{i+1,j-1}-U_{i-1,j+1}-U_{i-1,j-1}\right) - \right.\right.$$

$$-\frac{1}{4\Delta\xi\Delta\eta}\left(x_{i+1,j+1}-x_{i+1,j-1}-x_{i-1,j+1}-x_{i-1,j-1}\right)\frac{U_{i+1,j}-2U_{i,j}-U_{i-1,j}}{\Delta\xi^2}+\frac{1}{4\Delta\xi\Delta\eta}\left(y_{i+1,j+1}-y_{i+1,j-1}-y_{i-1,j+1}-y_{i-1,j-1}\right)$$

$$\frac{V_{i+1,j}-2V_{i,j}-V_{i-1,j}}{\Delta\xi^2}-\frac{y_{i+1,j}-2y_{i,j}-y_{i-1,j}}{\Delta\xi^2}\frac{1}{4\Delta\xi\Delta\eta}\left(V_{i+1,j+1}-V_{i+1,j-1}-V_{i-1,j+1}-V_{i-1,j-1}\right)\Bigg)\Bigg). \quad (3.55)$$

The discretized transformed momentum equation in the *y*-direction,

$$\left.\frac{\partial(Vh)}{\partial t}\right|_{i,j}^{k}=-\frac{1}{J}\left(\left.y_\eta\right|_{i,j}^{k}\left.\frac{\partial(\beta UVh)}{\partial\xi}\right|_{i,j}^{k}-\left.y_\xi\right|_{i,j}^{k}\left.\frac{\partial(\beta UVh)}{\partial\eta}\right|_{i,j}^{k}+\left.x_\xi\right|_{i,j}^{k}\left.\frac{\partial(\beta V^2h)}{\partial\eta}\right|_{i,j}^{k}-\left.x_\eta\right|_{i,j}^{k}\left.\frac{\partial(\beta V^2h)}{\partial\xi}\right|_{i,j}^{k}\right)-\left.fUh\right|_{i,j}^{k}$$

$$-\left.\frac{gh}{J}\right|_{i,j}^{k}\left(\left.x_\xi\right|_{i,j}^{k}\left.\frac{\partial\zeta}{\partial\eta}\right|_{i,j}^{k}-\left.x_\eta\right|_{i,j}^{k}\left.\frac{\partial\zeta}{\partial\xi}\right|_{i,j}^{k}\right)+\frac{\tau_{wy}-\tau_{by}}{\rho}+\frac{1}{\rho J}\left(\left.y_\eta\right|_{i,j}^{k}\left.\frac{\partial(hT_{xy})}{\partial\xi}\right|_{i,j}^{k}-\left.y_\xi\right|_{i,j}^{k}\left.\frac{\partial(hT_{xy})}{\partial\eta}\right|_{i,j}^{k}\right)+$$

$$+\frac{1}{\rho J}\left(\left.x_\xi\right|_{i,j}^{k}\left.\frac{\partial(hT_{yy})}{\partial\eta}\right|_{i,j}^{k}-\left.x_\eta\right|_{i,j}^{k}\left.\frac{\partial(hT_{yy})}{\partial\xi}\right|_{i,j}^{k}\right),$$

is

$$\left.\frac{\partial(Vh)}{\partial t}\right|_{i,j}^{k}=-\frac{1}{J}\left(\frac{y_{i,j+1}-y_{i,j-1}}{2\Delta\eta}\beta\frac{U_{i+1,j}V_{i+1,j}h_{i+1,j}-U_{i-1,j}V_{i-1,j}h_{i-1,j}}{2\Delta\xi}-\frac{y_{i+1,j}-y_{i-1,j}}{2\Delta\xi}\beta\frac{U_{i,j+1}V_{i,j+1}h_{i,j+1}-U_{i,j-1}V_{i,j-1}h_{i,j-1}}{2\Delta\eta}+\right.$$

$$\left.+\frac{x_{i+1,j}-x_{i-1,j}}{2\Delta\xi}\beta\frac{V^2{}_{i,j+1}h_{i,j+1}-V^2{}_{i,j-1}h_{i,j-1}}{2\Delta\eta}-\frac{x_{i,j+1}-x_{i,j-1}}{2\Delta\eta}\beta\frac{V^2{}_{i+1,j}h_{i+1,j}-V^2{}_{i-1,j}h_{i-1,j}}{2\Delta\xi}\right)-$$

$$-fU_{i,j}h_{i,j}-g\frac{h_{i,j}}{J}\left(\frac{x_{i+1,j}-x_{i-1,j}}{2\Delta\xi}\frac{\zeta_{i,j+1}-\zeta_{i,j-1}}{2\Delta\eta}-\frac{x_{i,j+1}-x_{i,j-1}}{2\Delta\eta}\frac{\zeta_{i+1,j}-\zeta_{i-1,j}}{2\Delta\xi}\right)+\frac{\tau_{wy}-\tau_{by}}{\rho}+$$

$$+\frac{1}{\rho J}\left(\frac{y_{i,j+1}-y_{i,j-1}}{2\Delta\eta}\frac{h_{i+1,j}-h_{i-1,j}}{2\Delta\xi}\frac{\rho\bar{\varepsilon}}{J}\left(\frac{x_{i+1,j}-2x_{i,j}-x_{i-1,j}}{\Delta\xi^2}\frac{1}{4\Delta\xi\Delta\eta}\left(U_{i+1,j+1}-U_{i+1,j-1}-U_{i-1,j+1}-U_{i-1,j-1}\right)-\right.\right.$$

$$-\frac{1}{4\Delta\xi\Delta\eta}\left(x_{i+1,j+1}-x_{i+1,j-1}-x_{i-1,j+1}-x_{i-1,j-1}\right)\frac{U_{i+1,j}-2U_{i,j}-U_{i-1,j}}{\Delta\xi^2}+\frac{1}{4\Delta\xi\Delta\eta}\left(y_{i+1,j+1}-y_{i+1,j-1}-y_{i-1,j+1}-y_{i-1,j-1}\right)$$

$$\left.\left.\frac{V_{i+1,j}-2V_{i,j}-V_{i-1,j}}{\Delta\xi^2}-\frac{y_{i+1,j}-2y_{i,j}-y_{i-1,j}}{\Delta\xi^2}\frac{1}{4\Delta\xi\Delta\eta}\left(y_{i+1,j+1}-y_{i+1,j-1}-y_{i-1,j+1}-y_{i-1,j-1}\right)\right)\right)$$

$$+\frac{1}{\rho J}\left(-\frac{y_{i+1,j}-y_{i-1,j}}{2\Delta\xi}\frac{h_{i,j+1}-h_{i,j-1}}{2\Delta\eta}\frac{\rho\bar{\varepsilon}}{J}\left(\frac{1}{4\Delta\xi\Delta\eta}\left(x_{i+1,j+1}-x_{i+1,j-1}-x_{i-1,j+1}-x_{i-1,j-1}\right)\frac{U_{i+1,j}-2U_{i,j}-U_{i-1,j}}{\Delta\xi^2}-\right.\right.$$

$$\frac{x_{i+1,j}-2x_{i,j}-x_{i-1,j}}{\Delta\xi^2}\frac{1}{4\Delta\xi\Delta\eta}\left(U_{i+1,j+1}-U_{i+1,j-1}-U_{i-1,j+1}-U_{i-1,j-1}\right)+\frac{y_{i+1,j}-2y_{i,j}-y_{i-1,j}}{\Delta\xi^2}$$

$$\frac{1}{4\Delta\xi\Delta\eta}\left(V_{i+1,j+1}-V_{i+1,j-1}-V_{i-1,j+1}-V_{i-1,j-1}\right)-\frac{1}{4\Delta\xi\Delta\eta}\left(y_{i+1,j+1}-y_{i+1,j-1}-y_{i-1,j+1}-y_{i-1,j-1}\right)\frac{V_{i,j+1}-2V_{i,j}-V_{i,j-1}}{\Delta\eta^2}\Bigg)\Bigg)+$$

$$+\frac{1}{\rho J}\Bigg(\frac{x_{i+1,j}-x_{i-1,j}}{2\Delta\xi}\frac{h_{i,j+1}-h_{i,j-1}}{2\Delta\eta}\frac{2\rho\bar{\varepsilon}}{J}\Bigg(\frac{1}{4\Delta\xi\Delta\eta}\left(x_{i+1,j+1}-x_{i+1,j-1}-x_{i-1,j+1}-x_{i-1,j-1}\right)\frac{V_{i,j+1}-2V_{i,j}-V_{i,j-1}}{\Delta\eta^2}-$$

$$-\frac{x_{i,j+1}-2x_{i,j}-x_{i,j-1}}{\Delta\eta^2}\frac{1}{4\Delta\xi\Delta\eta}\left(V_{i+1,j+1}-V_{i+1,j-1}-V_{i-1,j+1}-V_{i-1,j-1}\right)\Bigg)\Bigg)+$$

$$+\frac{1}{\rho J}\Bigg(-\frac{x_{i,j+1}-x_{i,j-1}}{2\Delta\eta}\frac{h_{i+1,j}-h_{i-1,j}}{2\Delta\xi}\frac{2\rho\bar{\varepsilon}}{J}\Bigg(\frac{x_{i+1,j}-2x_{i,j}-x_{i-1,j}}{\Delta\xi^2}\frac{1}{4\Delta\xi\Delta\eta}\left(V_{i+1,j+1}-V_{i+1,j-1}-V_{i-1,j+1}-V_{i-1,j-1}\right)-$$

$$-\frac{1}{4\Delta\xi\Delta\eta}\left(x_{i+1,j+1}-x_{i+1,j-1}-x_{i-1,j+1}-x_{i-1,j-1}\right)\frac{V_{i+1,j}-2V_{i,j}-V_{i-1,j}}{\Delta\xi^2}\Bigg)\Bigg). \qquad (3.56)$$

3.8 Boundary conditions for the numerical schemes

Figure 3.5 provides an illustrative sketch of the indexing system used in discretising the computational domain and its boundaries. At solid walls either slip or no-slip conditions are applied, as indicated below.

The solid wall slip condition is:

$$h\big|_{0,j}=2h\big|_{1,j}-h\big|_{2,j}\ ,\ \ U\big|_{0,j}=U\big|_{1,j}\text{, and } V\big|_{0,j}=0 \text{ at the west boundary}$$

$$h\big|_{i\max,j}=2h\big|_{i\max-1,j}-h\big|_{i\max-2,j}\ ,\ \ U\big|_{i\max,j}=U\big|_{i\max-1,j}\text{, and } V\big|_{i\max,j}=V\big|_{i\max-1,j} \text{ at the east boundary}$$

$$h\big|_{i,0}=2h\big|_{i,1}-h\big|_{i,2}\ ,\ U\big|_{i,0}=U\big|_{i,1}\ \text{, and }\ V\big|_{i,0}=0 \text{ at the south boundary}$$

and

$$h\big|_{i,j\max}=2h\big|_{i,j\max-1}-h\big|_{i,j\max-2}\ ,\ U_{i,j\max}=U\big|_{i,j\max-1}\ ,\ V\big|_{i,j\max}=0 \text{ at the north boundary}$$

The solid wall no-slip condition is:

$$h\big|_{0,j}=2h\big|_{1,j}-h\big|_{2,j}\ ,\ \ U\big|_{0,j}=0\text{, and } V\big|_{0,j}=0 \text{ at the west boundary}$$

$$h\big|_{i\max,j} = 2h\big|_{i\max-1,j} - h\big|_{i\max-2,j} \ , \ U\big|_{i\max,j} = 0 \text{, and } V\big|_{i\max,j} = 0 \text{ at the east boundary}$$

$$h\big|_{i,0} = 2h\big|_{i,1} - h\big|_{i,2} \text{ , } U\big|_{i,0} = 0 \text{ , and } V\big|_{i,0} = 0 \text{ at the south boundary}$$

and

$$h\big|_{i,j\max} = 2h\big|_{i,j\max-1} - h\big|_{i,j\max-2} \text{ , } U_{i,j\max} = 0 \text{ , } V\big|_{i,j\max} = 0 \text{ at the north boundary}$$

The open (extrapolation/transmissive) condition at the inflow Cartesian boundary is:

$$h\big|_{0,j} = 2h\big|_{1,j} - h\big|_{2,j} \ , \ U\big|_{0,j} = U\big|_{1,j} \text{, and } V\big|_{0,j} = V\big|_{1,j} \text{ at the west boundary}$$

$$h\big|_{i\max,j} = 2h\big|_{i\max-1,j} - h\big|_{i\max-2,j} \ , \ U\big|_{i\max,j} = U\big|_{i\max-1,j} \text{, and } V\big|_{i\max,j} = V\big|_{i\max-1,j} \text{ at the east boundary}$$

3.9. Adams-Bashforth numerical integration

To update the shallow water equations and complete the numerical scheme, second-order Adams-Bashforth time integration is employed, whereby a variable *f* at point *i, j* is marched forward one time step given knowledge of the temporal gradients in *f* at the present and previous time levels *k* and *k*-1 (Kreyszig, 2011, p911):

$$f\big|_{i,j}^{k+1} = f\big|_{i,j}^{k} + \Delta t\left(\frac{3}{2}\frac{df}{dt}\bigg|_{i,j}^{k} - \frac{1}{2}\frac{df}{dt}\bigg|_{i,j}^{k-1}\right).$$

Hence, the updated water depth estimated from the discretised continuity equation (equation 3.51 in the Cartesian system and equation 3.54 in the transformed system) is given by

$$h\big|_{i,j}^{k+1} = h\big|_{i,j}^{k} + \Delta t\left(\frac{3}{2}\frac{dh}{dt}\bigg|_{i,j}^{k} - \frac{1}{2}\frac{dh}{dt}\bigg|_{i,j}^{k-1}\right).$$

Similarly, the *x*-direction velocity component estimated from the relevant momentum equation (equation 3.52 in the Cartesian system and equation 3.55 in the transformed system) is

$$U\Big|_{i,j}^{k+1} = U\Big|_{i,j}^{k} + \Delta t\left(\frac{3}{2}\frac{dU}{dt}\Big|_{i,j}^{k} - \frac{1}{2}\frac{dU}{dt}\Big|_{i,j}^{k-1}\right).$$

And finally, the *y*-direction velocity component estimated from the corresponding momentum equation (equation 3.53 in the Cartesian system and equation 3.56 in the transformed system) is

$$V\Big|_{i,j}^{k+1} = V_{i,j}^{k} + \Delta t\left(\frac{3}{2}\frac{dV}{dt}\Big|_{i,j}^{k} - \frac{1}{2}\frac{dV}{dt}\Big|_{i,j}^{k-1}\right).$$

3.10. Runge-Kutta numerical integration

The 4th-order Runge-Kutta method (RK4) is a very efficient, accurate, multi-step method that is widely applied to solve differential equations. At each step, four auxiliary values K_1, K_2, K_3, K_4 and then the new value f_{n+1} is computed (where *f* refers to the dependent variable). The RK4 method extends the mid-point method applied by Adams and Bashforth to include values at the quarter points of the time interval. Unlike the Adams-Bashforth method, the RK4 method utilizes solely present information and intermediate estimates when projecting forward. RK4 is therefore more accurate and permits use of a longer time step (of the order 10 × larger) than the Adams-Bashforth second-order method. Here, the updated water depth and horizontal velocity components are estimated using the RK4 method from the discretised continuity and momentum equations as follows (kreyszig, 2011, p904):

$$h\Big|_{i,j}^{k+1} = h\Big|_{i,j}^{k} + \frac{1}{6}\Delta t\left(K_{11} + 2K_{12} + 2K_{13} + K_{14}\right),$$

$$U\Big|_{i,j}^{k+1} = \Big|_{i,j}^{k} + \frac{1}{6}\Delta t\left(K_{21} + 2K_{22} + 2K_{23} + K_{24}\right),$$

and

$$V\Big|_{i,j}^{k+1} = V\Big|_{i,j}^{k} + \frac{1}{6}\Delta t\left(K_{31} + 2K_{32} + 2K_{33} + K_{34}\right).$$

in which

$$K_{11} = \left[\frac{dh}{dt}(t_{i,j}, h_{i,j})\right] \quad , \quad K_{21} = \left[\frac{dU}{dt}(t_{i,j}, U_{i,j})\right] \quad , \quad K_{31} = \left[\frac{dV}{dt}(t_{i,j}, V_{i,j})\right]$$

$$K_{12} = \left[\frac{dh}{dt}\left(t_{i,j} + \frac{\Delta t}{2}, h_{i,j} + \frac{K_{11}}{2}\right)\right], K_{22} = \left[\frac{dU}{dt}\left(t_{i,j} + \frac{\Delta t}{2}, U_{i,j} + \frac{K_{21}}{2}\right)\right], K_{32} = \left[\frac{dV}{dt}\left(t_{i,j} + \frac{\Delta t}{2}, V_{i,j} + \frac{K_{31}}{2}\right)\right]$$

$$K_{13} = \left[\frac{dh}{dt}\left(t_{i,j} + \frac{\Delta t}{2}, h_{i,j} + \frac{K_{12}}{2}\right)\right], K_{23} = \left[\frac{dU}{dt}\left(t_{i,j} + \frac{\Delta t}{2}, U_{i,j} + \frac{K_{22}}{2}\right)\right], K_{33} = \left[\frac{dV}{dt}\left(t_{i,j} + \frac{\Delta t}{2}, V_{i,j} + \frac{K_{32}}{2}\right)\right]$$

$$K_{14} = \left[\frac{dh}{dt}(t_{i,j} + \Delta t, h_{i,j} + K_{13})\right], \; K_{24} = \left[\frac{dU}{dt}(t_{i,j} + \Delta t, U_{i,j} + K_{23})\right], \; K_{34} = \left[\frac{dV}{dt}(t_{i,j} + \Delta t, V_{i,j} + K_{33})\right].$$

3.11. Numerical procedure for solving the Cartesian shallow water equations

The computer program used to solve the 2-D shallow water equations is divided into four subroutines comprising input, calculation, update and output. The input subroutine reads in information from a steering file on the open channel properties and flow features such as length and width of channel, water depth, initial velocity at inlet, Reynolds number, eddy viscosity, wind and bed stress, water density, gravity acceleration, Chézy number, bed roughness, the distance between the bed and the horizontal datum, duration of time simulation, time step, and numbers of grid points in the Cartesian *x*- and y-directions. The calculation subroutine solves the discretized shallow water continuity, *x*-momentum and *y*-momentum equations in Cartesian system by applying second order Adams-Bashforth's numerical scheme or the 4th order Runge-Kutta numerical scheme and appropriate boundary conditions. The update subroutine replaces the present values with new values of *h*, *U*, and *V* after each time integration. The output subroutine records the final results of water depth and velocity components in *x* and *y* directions.

3.12. Numerical procedure for solving the Shallow water equations in Transformed systems

To deal with rivers with arbitrarily curved geometries, a numerical solver of the curvilinear (transformed) shallow water equations has been developed from scratch. The structure of program is essentially the same as that for the Cartesian

system, with separate subroutines for input, calculation, update, and output. The input subroutine reads in the physical parameters of the river channel, and details of the curvilinear grid previously generated by solving the discretised partial differential equations for grid generation using either successive over-relaxation, multi-grid, or conjugate gradient methods, the choice of method specific to the problem under consideration. Chapter 2 provides a detailed description of the numerical methods used for grid generation, and their verification. The hydrodynamic calculation subroutine comprises numerical integration (using either 2^{nd}-order AB or 4^{th} order RK methods) of the transformed version of the shallow water continuity, *x*-momentum, and *y*-momentum equations to provide new values of water depth and velocity components with time. The update and output subroutines are exactly the same as for the Cartesian scheme. For each run the CPU time is also computed and output.

3.13. Conclusions

Chapter 3 has outlined the second-order numerical model used to solve the non-linear shallow water equations and predict the shallow river flow hydrodynamics. Both Cartesian and transformed (curvilinear) versions of the shallow water equations are solved numerically. The programs having been written from scratch by the author. Chapter 4 will describe the verification of the shallow water models to flow in open channels and wide rivers.

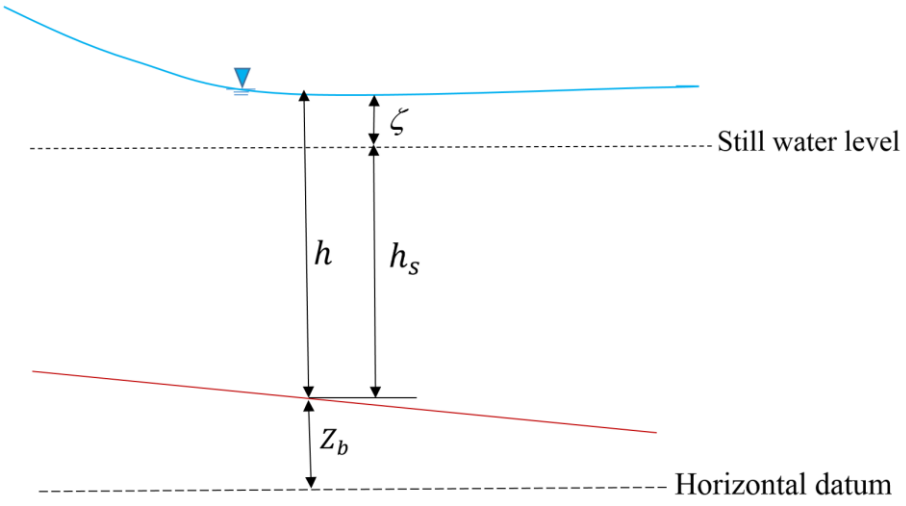


Figure 3.1 Shallow flow in an open channel

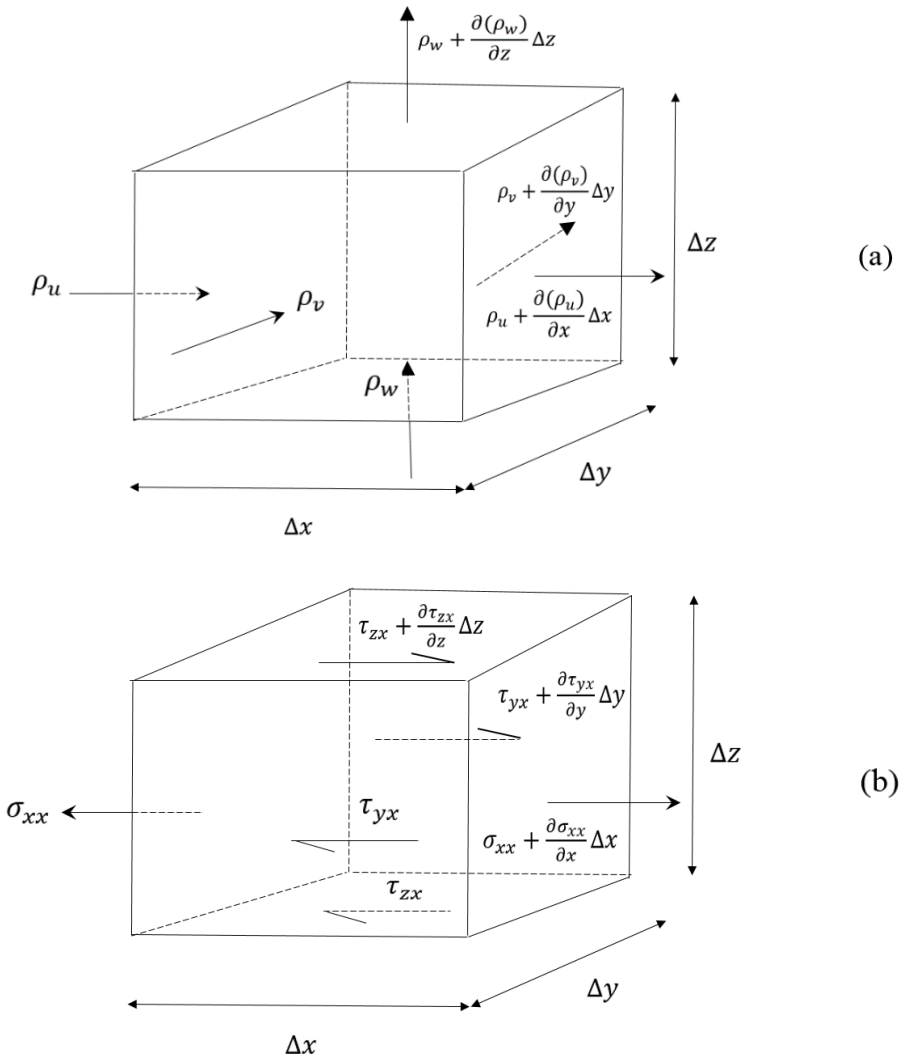


Figure 3.2(a) 3D inflow and outflow element. (b) Fluid stress components acting in the *x*-direction

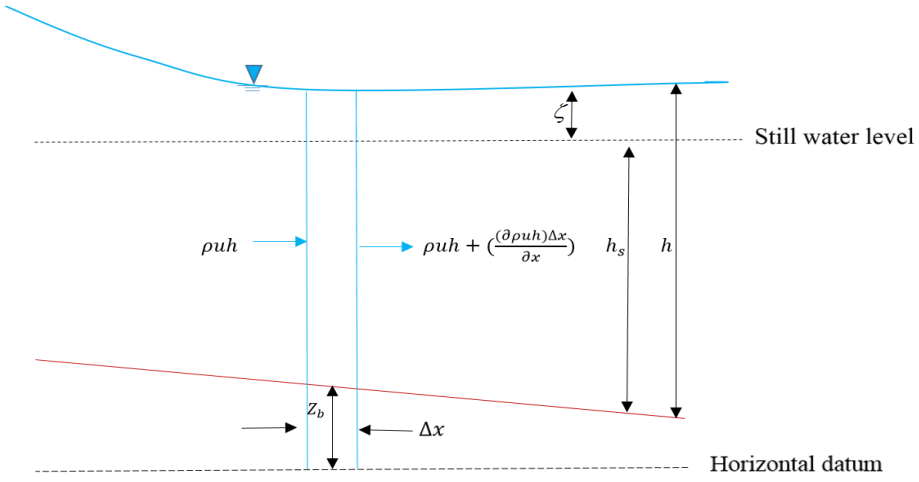


Figure 3.3 Elemental volume for continuity equation

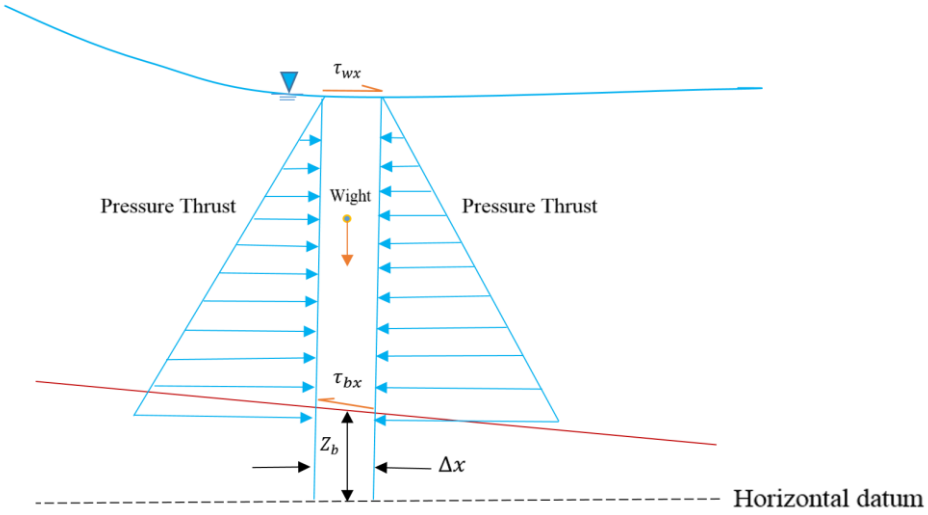


Figure 3.4 Elemental volume for 1-D horizontal momentum equation

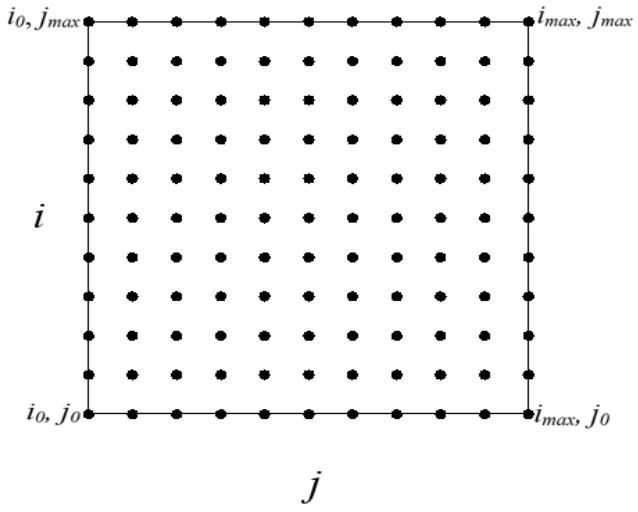


Figure 3.5 Schematic diagram of computational grid

Chapter4

Verification of Shallow Water Equation Solvers

This chapter deals with the verification of the numerical models based on the Cartesian and the curvilinear shallow water equations developed as part of this thesis to simulate environmental free surface flows in wide rivers. Results will be presented for four test cases: uniform flow in an open channel; wind induced set-up in rectangular and circular basins, recirculation in an open channel with a side wall expansion; and shallow flow in a rectangular channel with single groyne. In all cases, the density of water is 1000 kg/m^3, and the acceleration due to gravity is 9.81 m/s^2.

4.1 Uniform flow in an open channel

4.1.1 Cartesian grid results

The benchmark case of steady uniform flow in a sloping rectangular channel is used to verify that the shallow flow solver provides the correct balance between flow gradient and bed resistance terms. Local and convective acceleration terms are also tested as the flow builds up from initially still conditions. The open channel has length 1000 m and width 240 m. Its lateral walls are frictionless. Bed friction is represented by means of the Chézy coefficient, C = 40 m$^{1/2}$s^{-1}. The eddy viscosity is set to zero. The initial water depth is 5 m throughout the channel, bed slope is 1:1000, and the initial mass flux is set to zero everywhere. The bed elevation is given by

$$z_b|_{i,j} = 2 - \frac{x|_{i,j}}{L_x} \quad , \qquad (4.1)$$

in which $x|_{i,j}$ is the grid location in x-direction, and L_x = 1000 m is the length of the channel.

Here, the grid is rectangular with 100 cells in the stream-wise *x*-direction and 20 cells in the transverse *y*-direction, as shown in Figure 4.1. Two time integration methods are considered: the 2nd-order Adams-Bashforth (AB) scheme with time step set to 1.0 s; and the 4th-order Runge-Kutta (RK4) scheme with time step of 10 s. Figure 4.2 shows the close agreement in the velocity time histories in the channel obtained using the two time integration schemes in the Cartesian SWEs solver. Steady state uniform flow conditions are reached by time t = 2000 s. Figure 4.3 indicates the corresponding velocity time history obtained using the curvilinear SWEs solver. The results are the same as in Figure 4.2, as would be expected given that grid has remained Cartesian even if the solver is for the curvilinear shallow water equations. In terms of computational performance, the RK4 time integration gave stable, converged results much faster than AB time integration. In all cases, the flow velocity in the channel converged to a steady value, $U_{numerical} = 2.82842712$ m/s. For verification purposes the numerical prediction is compared against the analytical solution obtained by balancing the bed resistance to the weight component of the flow down the sloping channel. Using the Chézy friction law:

$$u_{Analytical} = C\sqrt{h\,S} = 40\sqrt{5\,\frac{1}{1000}} = 2.82842712\ \text{m/s} \qquad (4.2)$$

In the above equation, h is the water depth, and S is the channel bed slope. Also Froude number is:

$$Fr = \sqrt{\frac{U}{g\,h}} = \sqrt{\frac{2.82842712}{49.05}} = 0.101936799 \qquad (4.3)$$

There is almost perfect agreement between the analytical solution and the numerical predictions indicating that the numerical solver correctly models the balance between bed friction and flow gradient source terms, and that the advection processes are properly modelled as the flow tends to uniform conditions from rest. Also, the relative error is zero.

4.1.2 Distorted curvilinear grid results

To investigate how well the curvilinear shallow flow solver models the same case as above, but on a distorted mesh, the multi-grid (MG) program is used to create the physical and transformed computational grids shown in Figure 4.4. Here the physical grid has been created using a combination of single line attraction on one side, single line attraction on both sides, and single point attraction. Full details are given in Section 2.5. The channel dimensions and flow parameters are exactly the same as in Section 4.1. Figure 4.5 shows the velocity time history obtained at the centre of the channel using the curvilinear SWEs solver with a time step of 0.1 s. Here, the velocity saturates at 2.82842712 m/s and the relative error is zero, the same result previously obtained using the Cartesian SWEs solver. Figure 4.6 shows the steady state velocity field, in which all the vectors are aligned in the stream-wise direction (even though there is an optical effect caused by the vectors being plotted on the coordinates of the distorted grid). Very similar results were obtained by Borthwick and Barber (1992) using the successive-over relaxation (SOR) method to generate the grid for their shallow water solver. This study confirms the ability of the curvilinear shallow flow solver to predict uniform flow correctly on a distorted grid.

4.2 Wind-induced set up in rectangular basin

This test examines wind-induced set-up in a rectangular basin, and tests the balance between the surface wind stress and the surface elevation gradient terms in the governing equations. The model illustrates the application of viscous forces in shallow water equations. The rectangular basin is 10 km long and 10 km wide, the initial water depth is 10 m, wind stress in *x*-direction is τ_{wx}= 0.1 N/m^2, wind stress in *y*-direction is τ_{wy}= 0 N/m^2, bed roughness is *cf* = 100, eddy viscosity (υ) is 10 m^2/s, the elevation of the channel bed (z_b) is 2 m above a fixed horizontal datum, and basin has a flat bed. In this case, the effective stress components, T_{xx}, T_{xy} and T_{yy}, in the momentum equations have non-zero values at the start of the simulation because of the presence of velocity gradients, and the non-zero eddy viscosity. Energy is also dissipated through the bed frictional resistance associated with the high roughness coefficient. In

this case, the curvilinear shallow water equations are solved on a rectangular grid comprising 40 x 40 cells, using the 2nd-order Adams-Bashforth scheme with a time of 5 s and the 4th-order Runge-Kutta scheme with a time step of 50 s. The simulation lasts 80000 s. Figures 4.7 and 4.8 show the time evolution of the water depth and velocity at selected locations within the domain. It can be seen that steady state is reached by a time of about 2 x 10^4 s. It can be seen that the velocity time series predicted by the Adams-Bashforth scheme contain fluctuations, unlike those of the Runge-Kutta scheme that converges more quickly to steady state. According to the Adams-Bashforth scheme, the steady-state water depth at the left bank is 9.98993 m whereas that at the right bank is 10.0013 m. The set-up is the difference between depths either end of the basin, and is equal to 0.01137 m. Similarly, the Runge-Kutta scheme predicts steady-state water depths of 9.99490 m and 10.00509 m at the left and right ends of the basin, corresponding to a set-up of 0.01019 m. Similar to Barber's study (1990), in the analytical solution, the researcher has used the force balance method to determine the surface elevation gradient:

$$\frac{\partial \zeta}{\partial x} = \frac{\tau_{wx}}{\rho g h_0} \qquad or \qquad \frac{\Delta \zeta}{L} = \frac{\tau_{wx}}{\rho g h_0} \quad . \tag{4.4}$$

giving

$$\Delta \zeta \; = \frac{\tau_{wx} L}{\rho g h_0} \; = \frac{0.1 \times 10000}{1000 \times 9.81 \times 10} = 0.01019 \text{ m} \quad . \tag{4.5}$$

Both the Adams-Bathforth and the Runge-Kutta schemes give predicted values of set-up that are in satisfactory agreement with the analytical solution; the Runga-Kutta scheme is faster to run, does not generate oscillations in the velocity field, and is more accurate. Figure 4.9 displays the numerically predicted steady-state depth contours and a surface visualisation of the wind-induced set-up in the rectangular basin obtained using the Runge-Kutta scheme. The contours are straight and parallel, indicating that the set-up is uniform across the basin. This test confirms the ability of the solver to handle wind stresses, and the use of viscous stress terms to settle local acceleration and convective accelerations. The test also confirms that the Runge-Kutta scheme is superior to the 2nd order Adams-Bashforth scheme in predicting wind-induced set-up, in terms of both accuracy and computational performance.

The relative error of the analytical solution (exact solution) and Adams-Bathforth numerical scheme (approximate solution) is 0.115. However, the relative error of the Runge-Kutta scheme is zero. This indicates the accuracy of Runge-Kutta scheme.

4.3 Wind-induced set up in circular basin

This benchmark tests demonstrates the ability of the curvilinear shallow flow solver to predict wind-induced set-up in a curved domain. We consider a flat-bottomed circular basin of diameter 10 km, water depth 10 m, over which a wind stress of 0.1 N/m^2 is uniformly applied in *x* direction. The bed roughness is set so that *cf* = 100, and the eddy viscosity is 10 m^2/s. The bed elevation is 2 m above a fixed horizontal datum. Figure 4.10 shows the physical and transformed computational 20 x 20 cell grids obtained for this case, using the conjugate gradient method. Again, the 2nd-order Adams-Bashforth and 4th-order Runge-Kutta time integration schemes are used, with the time step set respectively to 1 and 10 s. In this case, the analytical solution for the set-up over the diameter of the basin is given by:

$$\Delta\zeta = \frac{\tau_{wx} D}{\rho g h_0} = \frac{0.1 \times 10000}{1000 \times 9.81 \times 10} = 0.01019 \text{ m} \quad . \qquad (4.6)$$

Figures 4.11 and 4.12 depict the evolution of the water level and horizontal velocity component at three locations in the basin with time, obtained using both time-integration schemes. Steady state appears to be reached by 4 x 10^4 s, with the 4th-order Runge-Kutta scheme giving very similar results to those of the 2nd-order Adams-Bashforth scheme, but at considerably slower computational speed (owing to the much smaller time step requirement). In this case, the predicted values of set up obtained using the 2nd-order Adams-Bashforth and 4th-order Runge-Kutta schemes are 0.0097 and 0.0113 m respectively, the latter value slightly closer to the analytical solution. Figure 4.13 shows the predicted contours of water depth in the basin. This study shows, the curvilinear shallow flow solver is capable of accurately simulating wind-induced set-up in a curved domain.

4.4 Flow in an open channel with a side wall expansion

This next test examines the implementation of effective stresses in the numerical model, and tests the convective terms for a case where the flow separates to form a recirculation zone in an open channel containing a side wall expansion. Figure 4.14 illustrates the open channel domain in plan. Many researchers have examined laminar flow over a step or flow in an open channel past a side-wall expansion. For open channel flow, it is usual to assume that the flow is laminar provided the inlet Reynolds number is sufficiently low, such that $Re_I \leq 2000$ where

$$Re_I = \frac{U_I b}{\upsilon} \qquad (4.7)$$

in which U_I is the inlet mean flow velocity through the inlet, b is half the width of the inlet and υ is the kinematic viscosity coefficient. Rogers (2001) solved the shallow water equations on an adaptive Cartesian quadtree grid using a Godunov-type finite volume scheme with Roe's Riemann solver, and presented velocity fields and recirculation lengths for Re_I = 7.9, 98 and 229. Rogers obtained very good agreement between his numerical results and previously published experimental data on air flow over a step, collected by Denham and Patrick (1974). It should be noted that the velocity profile across the inlet is assumed to be fully developed and have a parabolic distribution, such that

$$U(y) = \frac{3}{2} U_I \left(1 - \left(\frac{y}{b}\right)^2\right) \qquad (4.8)$$

where $U(y)$ is the local velocity component in the x-direction at any y-location across the inlet, and y has its origin midway across the inlet and is oriented in the transverse direction. It should be noted that the maximum value of inlet velocity occurs as the centre of the channel where $y = 0$, such that:

$$U_{max} = \frac{3}{2} U_I \qquad (4.9)$$

In the present study, shallow flow in an open channel with a side wall expansion is simulated, and the resulting recirculation lengths compared to Denham and Patrick's (1974) experimental data and Rogers' (2001) numerical predictions. The channel has a total length of L_x = 18 m and its upstream width is $2b$ = 2 m. At the side-

wall expansion the channel width alters to 3 m. The initial water depth is 1 m throughout. The bed friction coefficient and wind stress components are set to zero. The channel bed is completely horizontal, and located an elevation 2 m above a fixed horizontal datum (such that z_b = 2 m). A no-slip condition is applied at side walls. At the inlet open boundary, the inflow velocity is prescribed such that at steady-state the value is U_I = 0.5 m/s, and a transmissive condition used for the depth. The inlet velocity is ramped up from zero using a tanh function over a period of 80 s. The depth is fixed at the outlet boundary to 1 m, and the velocity determined by extrapolation. Laminar flows at inlet Reynolds numbers Re_I = 7.9, 73, 98, and 229 were simulated by setting the eddy viscosity to υ = 0.0633, 0.00685, 0.0051 and 0.00218 m^2/s respectively. Figure 4.15 shows the 360 x 60 Cartesian mesh on which the flow simulations were undertaken using the curvilinear shallow flow solver with 4^{th}-order Runge-Kutta time integration. The time step is 0.01 s. Figure 4.16 shows the steady state velocity vectors and stream function contours downstream of the expansion for Re_I = 7.9, 98, and 229. Also Figure 4.17 presents the velocity magnitude profiles downstream of the side wall expansion. It is clear from the figures that the length of the recirculation zone increases progressively with Re_I. Table 4.1 compares the results obtained by O'Leary and Muller, and Rogers with those of the current study.

4.5 Shallow flow in a rectangular channel with single groyne

Figure 4.18 is a definition sketch showing a plan view of the rectangular channel with single groyne. The length and width of the channel are *Lx* and *Ly* respectively. For the tests cases considered herein, *Lx* = 24 m, *Ly* =3 m, and the permeable groyne is represented by a 1 x 1 m patch of increased roughness, such that *cf* = 100. The roughness elsewhere is zero. In this study, the bed roughness provides the flow separation and develops the recirculation zone in the vicinity of groyne, however, the flow can pass through the permeable groyne. Table 4.2 lists the model parameters. A prescribed parabolic velocity profile is applied at the inlet, an extrapolation open boundary at the outlet, and no-slip conditions at the lateral walls. Here, eddy viscosities are 0.0633, 0.0051 m^2/s, and 0.00218 m^2/s corresponding to inlet Reynolds numbers of Re_I = 7.9, 98 and 229. Figure 4.19 shows the velocity vectors and

streamlines obtained for Re_I = 229 on two Cartesian computational grids, one comprising 360 x 60 cells, the other 720 x 120 cells. Each simulation was run for 80s.

The results are in satisfactory agreement indicating that the grid convergence is being achieved. Figures 4.20, 4.21 and 4.22 depict the streamlines and velocity vectors obtained at steady state for Re_I = 7.9, 98 and 229. We can see the growth in length of the recirculation zone behind the groyne as the Reynolds number increases. At Re_I = 229, there is also a small vortex discernible in front of the groyne. The results are in broad agreement with those for flow past a side wall expansion, though it must be noted that the inlet Reynolds numbers are based on different inlet flow widths (for the side-wall expansion, the inlet flow width is narrower than that for the groyne case). This test indicates that the use of artificial roughness is acceptable in mimicking the effect of a solid structure, such as a groyne. Here, almost no flow passes through the structure; instead it bypasses the groyne with a fast flow in the narrowest section followed by partial recovery of the flow profile far downstream.

4.6 Conclusion

This chapter has considered four case studies that test the ability of the Cartesian and curvilinear shallow flow solvers to predict open channel flow over a rough boundary, wind-induced set-up in the rectangular and circular basins, recirculating flow behind a side-wall expansion, and Shallow flow in a rectangular channel with single groyne. It is found that the curvilinear version of the code works well, by default on Cartesian grids. It is also confirmed that the Runge-Kutta time integration scheme is superior to the Adams-Bashforth time integrator, and so is used for all the shallow flow test cases considered in the following chapters.

Table 4.1. Length of the recirculation zone in an open channel with a side wall expansion

O'Leary and Muller, experimental results (*Re*=7.9)	O'Leary and Muller experimental results (*Re*=98)	O'Leary and Muller experimental results (*Re*=229)
1.5*b*	5.6*b*	unpredicted
Rogers, numerical Cartesian quadtree grid and SWEs for (*Re*=7.9)	Rogers numerical Cartesian quadtree grid and SWEs for (*Re*=98)	Rogers numerical Cartesian quadtree grid and SWEs for (*Re*=229)
1.29*b*	5.4*b*	9.6*b*
Jalali, curvilinear shallow flow solver (*Re*=7.9)	Jalali, curvilinear shallow flow solver (*Re*=98)	Jalali, curvilinear shallow flow solver (*Re*=229)
1.43*b*	5.57*b*	6.3*b*

Table 4.2. Channel properties for the rectangular channel with single groyne

L_x (m)	L_y (m)	h_0 (m)	Chézy ($m^{1/2}s^{-1}$)	z_b (m)	S	ρ (kg/m^3)	g (m^2/s)
24.0	3.0	1.0	0.0	2.0	0.0	1000.0	9.81

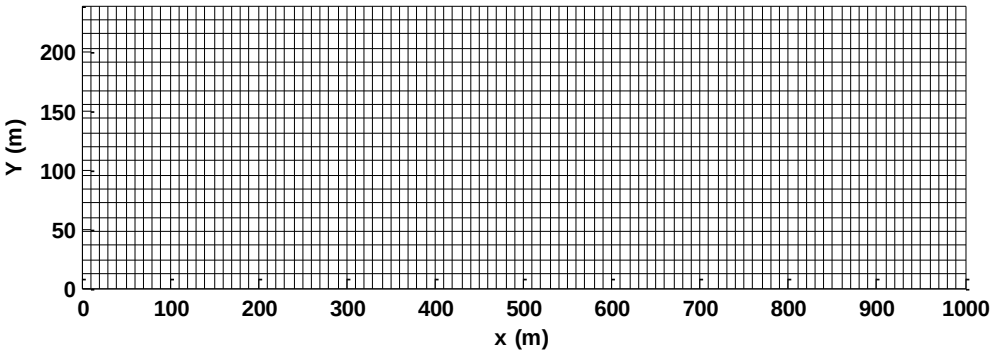
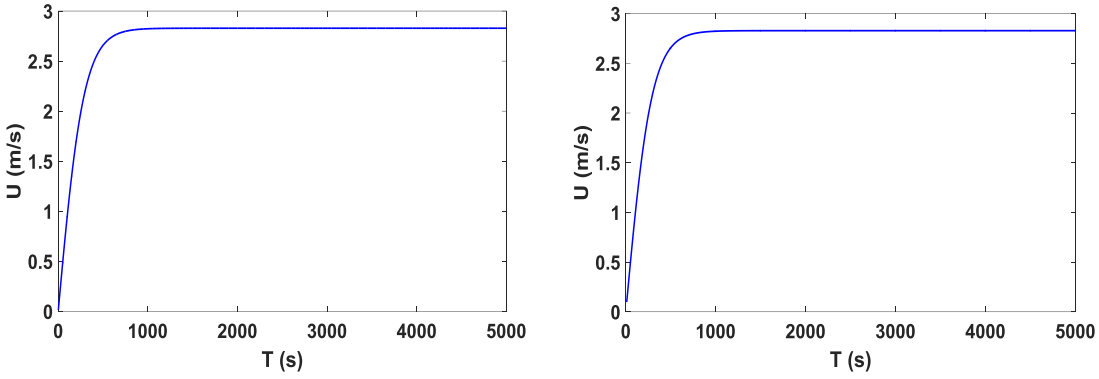
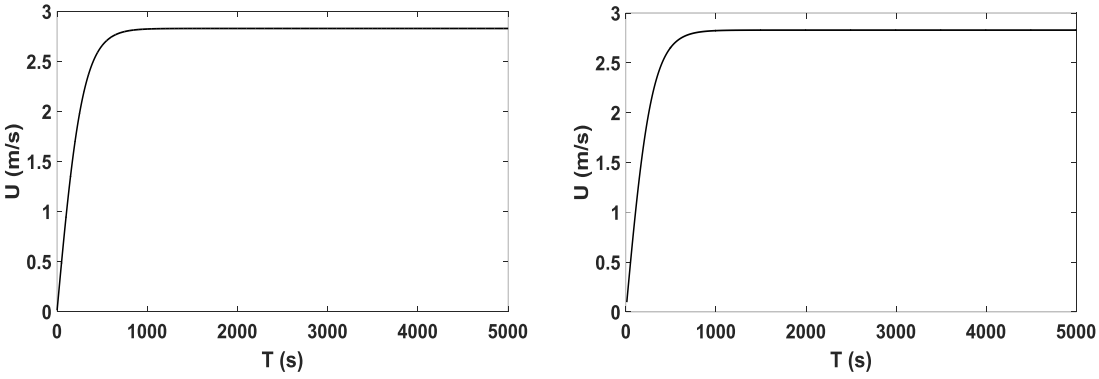


Figure 4.1 Cartesian grid for uniform flow in a rectangular channel.



(a) (b)

Figure 4.2 Velocity time histories for uniform flow in an open channel on rectangular grid, using Cartesian SWEs solver: (a) 2nd-order Adams-Bashforth time integration with $\Delta t = 1.0s$; (b) 4th-order Runge-Kutta time integration with $\Delta t = 10.0s$



(a) (b)

Figure 4.3 Velocity time histories for uniform flow in an open channel on rectangular grid, using curvilinear SWEs solver: (a) 2nd-order Adams-Bashforth time integration with $\Delta t = 1.0s$; (b) 4th-order Runge-Kutta time integration with $\Delta t = 10.0s$

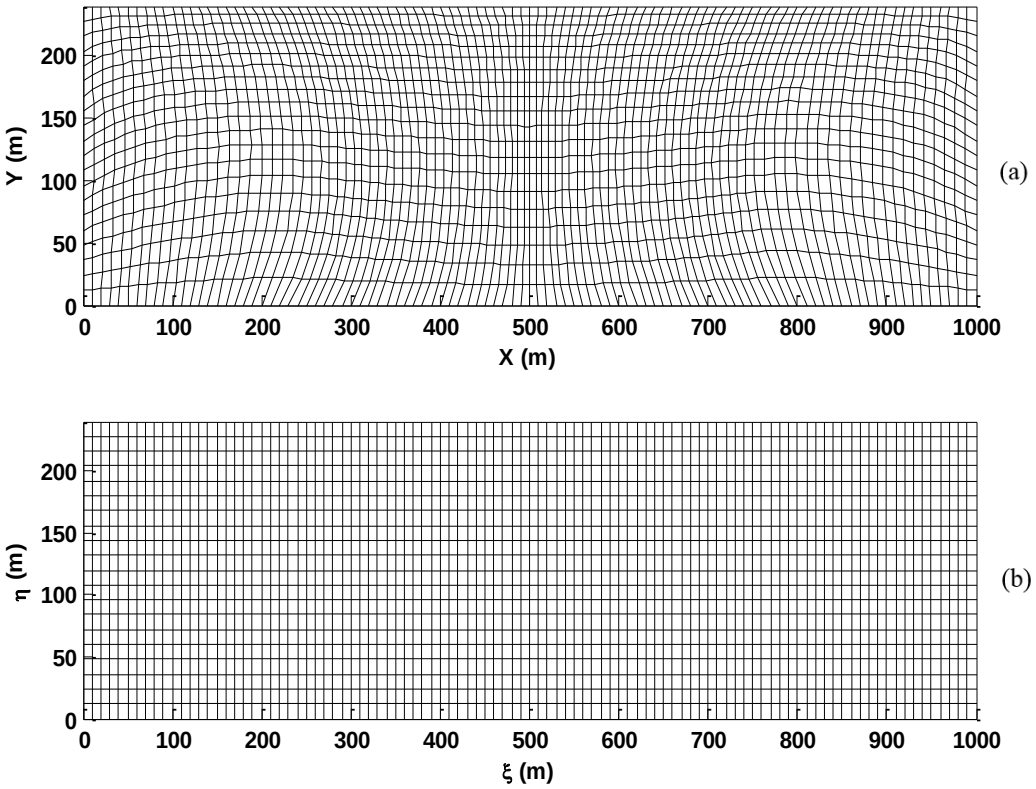


Figure 4.4 Uniform flow in an open channel: (a) physical grid and (b) transformed grid obtained using MG.

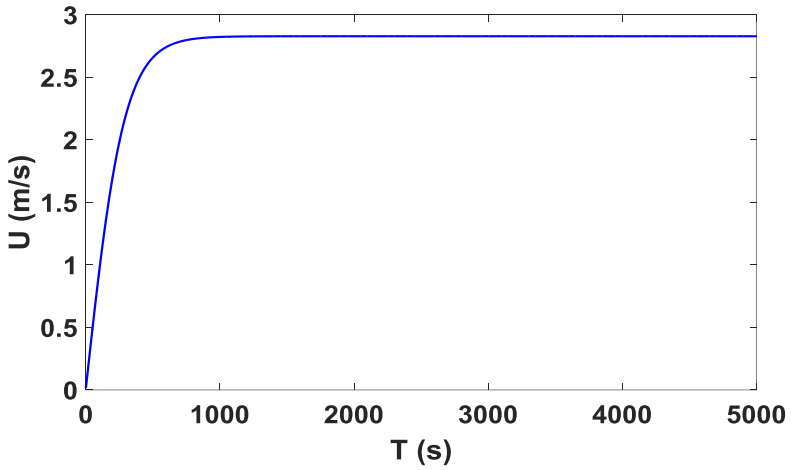


Figure 4.5 Velocity time histories for uniform flow at a point in the middle of an open channel, using the curvilinear SWEs solver on a distorted grid with 2nd-order Adams-Bashforth time integration and $\Delta t = 0.1s$

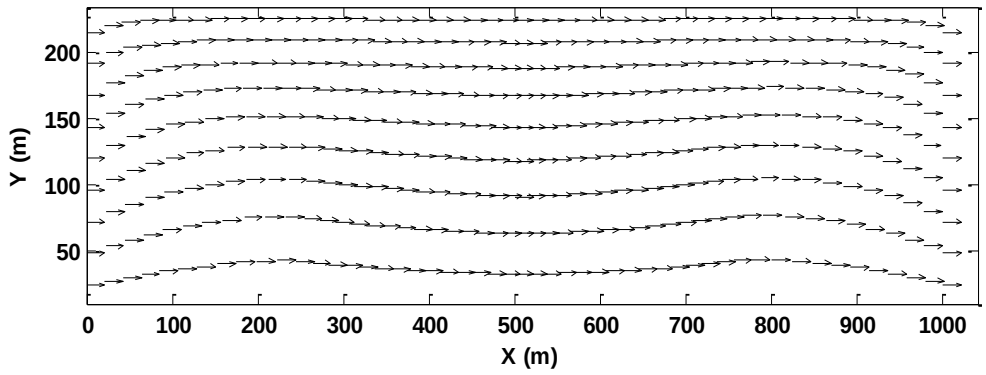


Figure 4.6 Steady state velocity field obtained for uniform flow in an open channel using curvilinear SWEs solver on distorted grid generated using MG iteration.

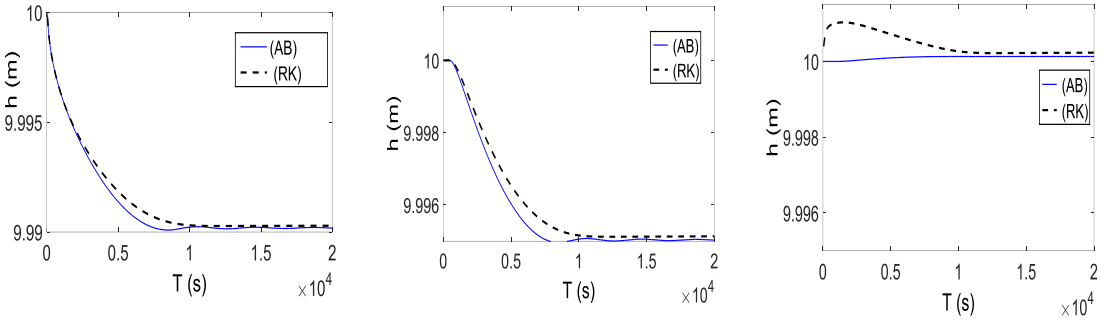


Figure 4.7 Wind-induced set-up in a rectangular basin of length 10 km, initial depth 10 m, subject to a uniform wind stress of 0.1 N/m^2, predicted using curvilinear SWEs solver with AB2 and RK4 time-integration. Water depth time histories at: (a) left bank; (b) centre; and (c) right bank.

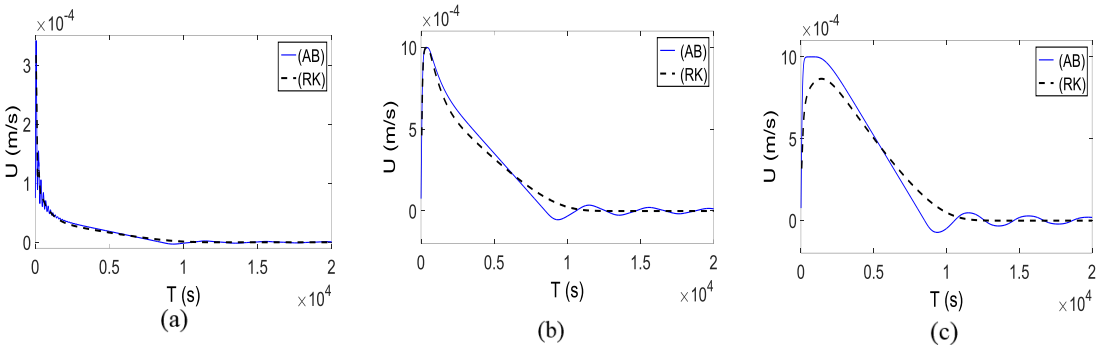


Figure 4.8 Wind-induced set-up in a rectangular basin of length 10 km, initial depth 10 m, subject to a uniform wind stress of 0.1 N/m^2, predicted using curvilinear SWEs solver with AB2 and RK4 time-integration. Horizontal velocity time histories at: (a) left bank; (b) centre; and (c) right bank.

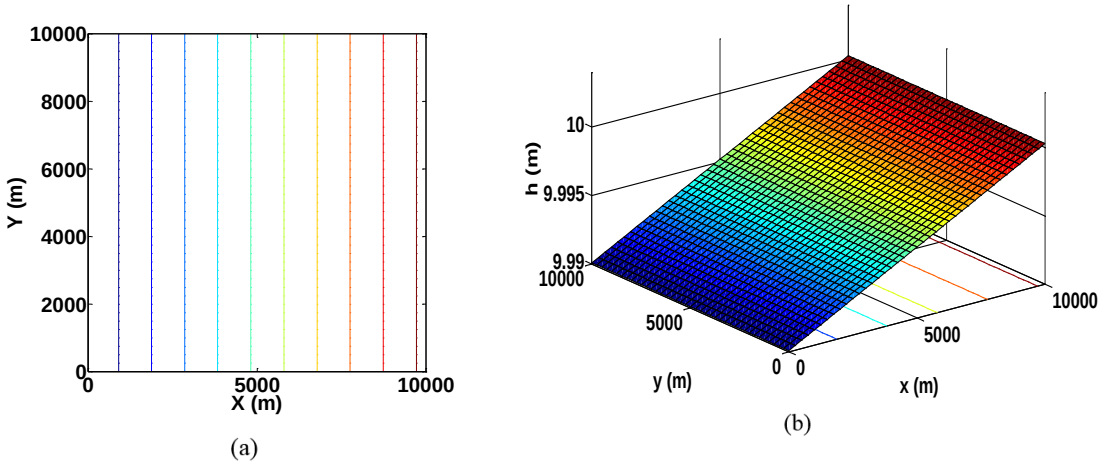


Figure 4.9 Wind-induced set-up in a rectangular basin of length 10 km, initial depth 10 m, subject to a uniform wind stress of 0.1 N/m^2, predicted using curvilinear SWEs solver with RK4 time-integration: (a) depth-contours; and (b) depth profile.

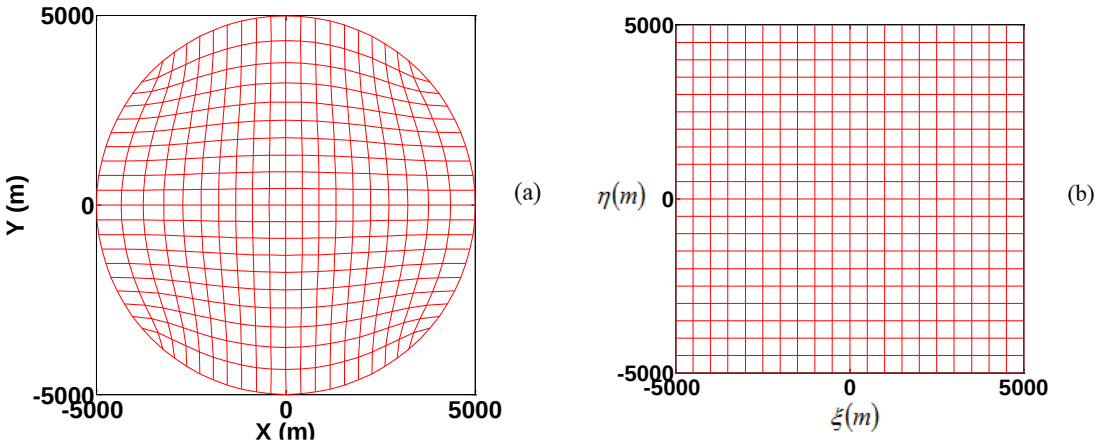


Figure 4.10 Wind-induced set-up in a circular basin: (a) physical grid and (b) transformed grid obtained using CG.

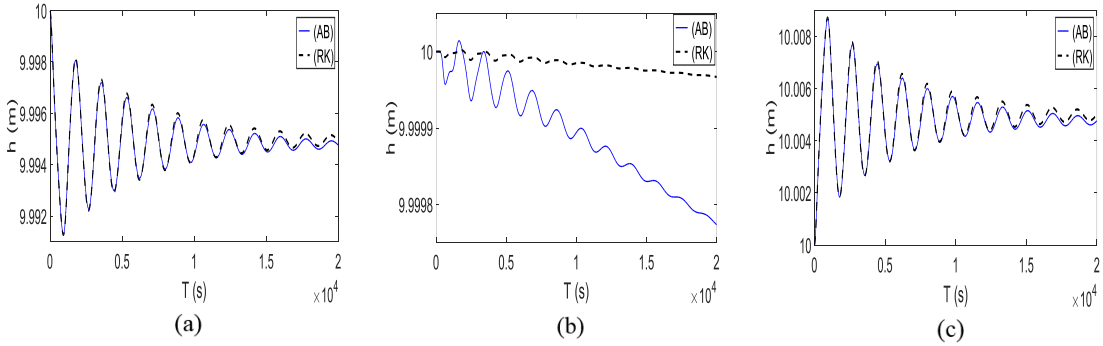


Figure 4.11 Wind-induced set-up in a circular basin of diameter 10 km, initial depth 10 m, subject to a uniform wind stress of 0.1 N/m^2, predicted using curvilinear SWEs solver with RK4 time-integration. Water depth time histories: (a) left bank; (b) centre; and (c) right bank.

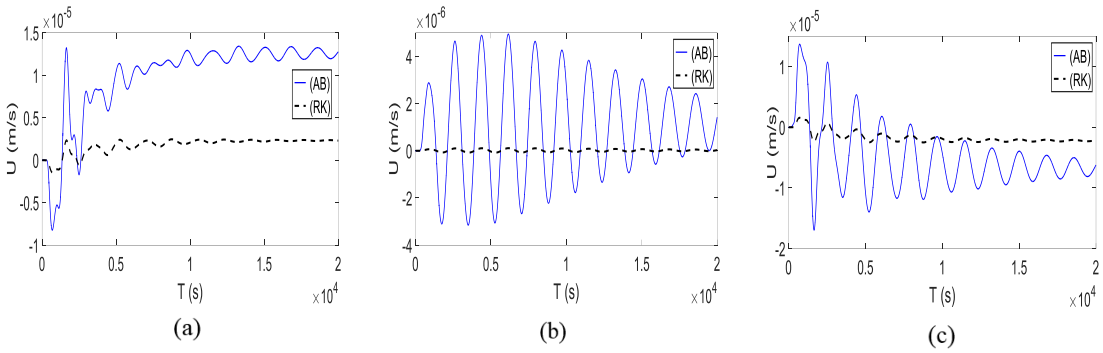


Figure 4.12 Wind-induced set-up in a circular basin of diameter 10 km, initial depth 10 m, subject to a uniform wind stress of 0.1 N/m^2, predicted using curvilinear SWEs solver with RK4 time-integration. Velocity time histories: (a) left bank; (b) centre; and (c) right bank.

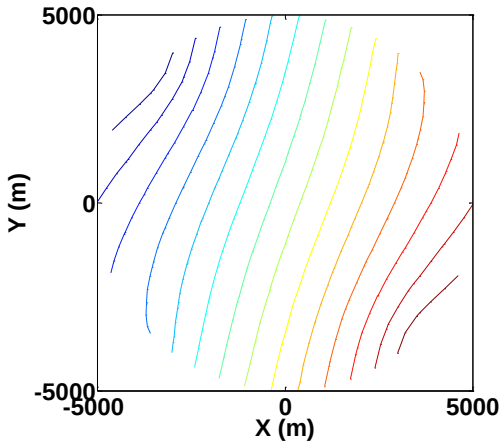


Figure 4.13 Steady-state depth-contours indicating wind-induced set-up in a circular basin of diameter 10 km, initial depth 10 m, subject to a uniform wind stress of 0.1 N/m^2, predicted using curvilinear SWEs solver with RK4 time-integration.

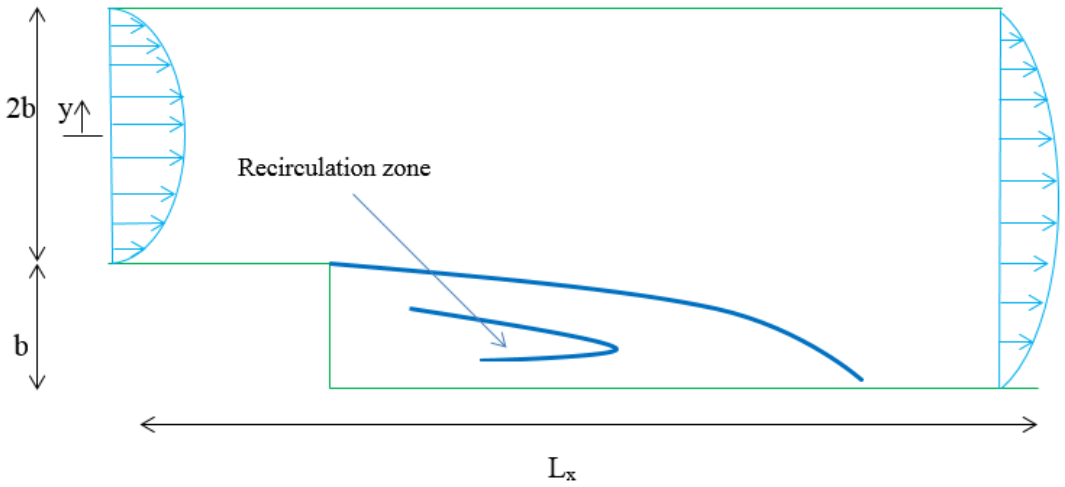


Figure 4.14 Definition sketch for an open channel containing a side wall expansion.

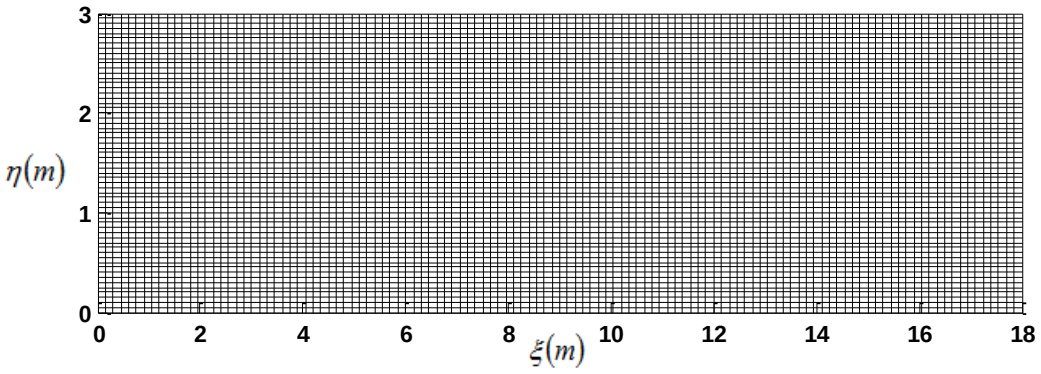


Figure 4.15 Curvilinear model for Channel with side wall expansion with 360 x 60 grids.

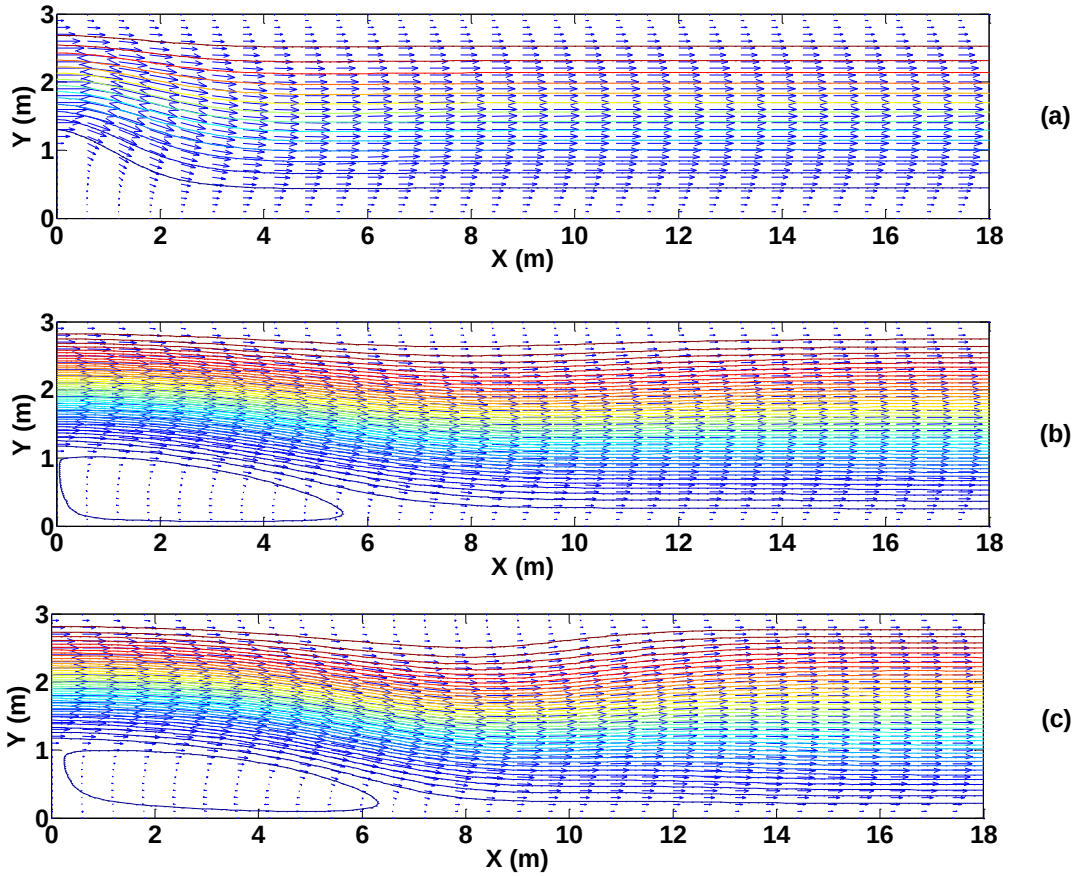


Figure 4.16 Side wall expansion in open channel: velocity vectors and stream function contours downstream of the expansion for: (a) Re_I = 7.9, (b) Re_I = 98; and (c) Re_I = 229.

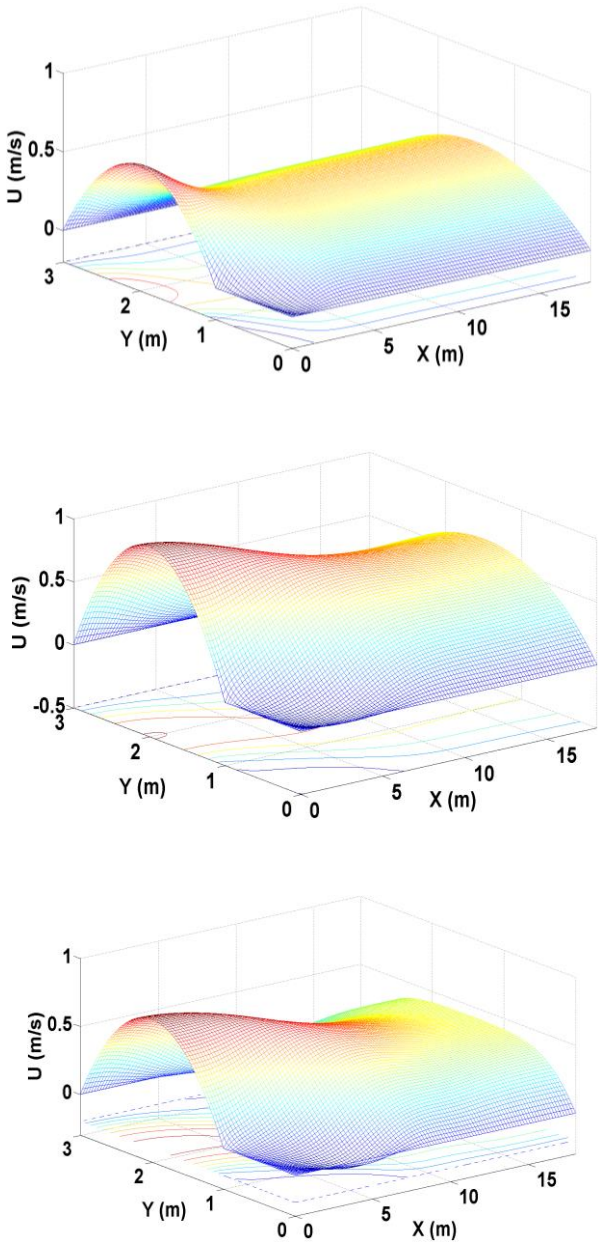


Figure 4.17 Velocity magnitude distributions downstream of the side wall expansion: (a) Re_I = 7.9, (b) Re_I = 98; and (c) Re_I = 229.

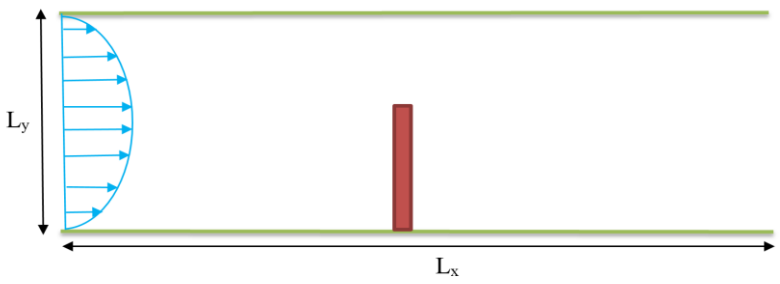
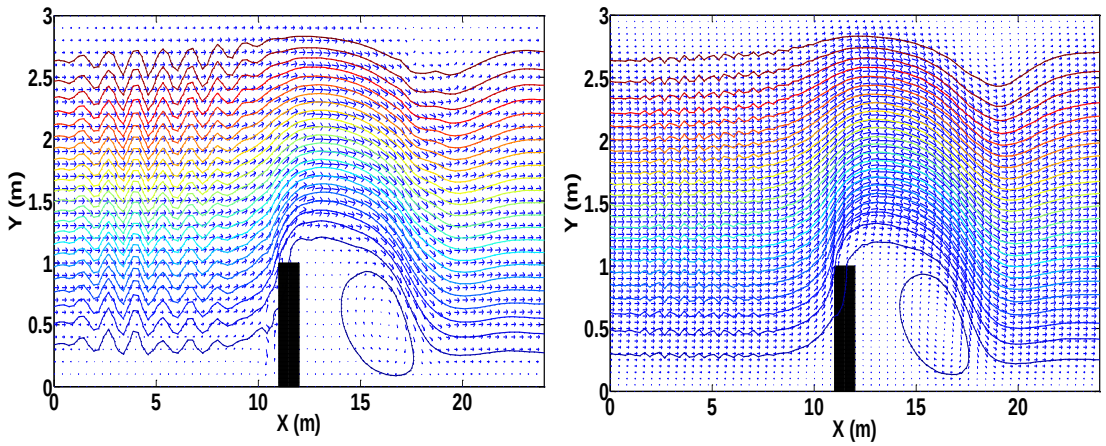
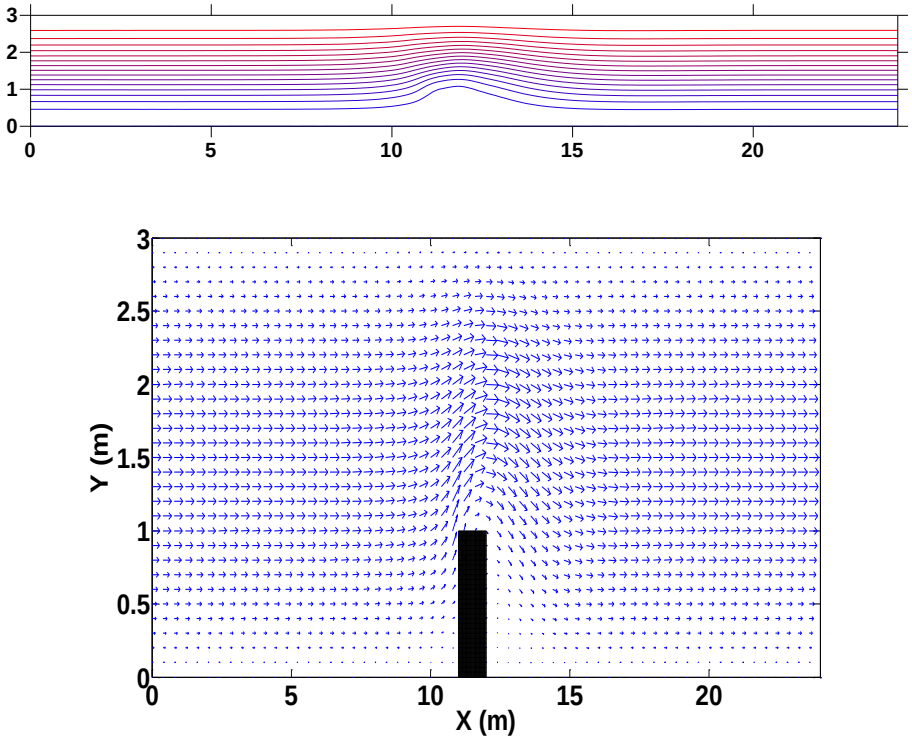


Figure 4.18 Rectangular channel with single groyne.



(a) (b)

Figure 4.19 Flow past an idealised groyne for Re_I = 229 at t = 80 s computed using the curvilinear grid code. Grid convergence test results showing stream function contours and velocity vectors on (a) 360 x 60 mesh; and (b) 720 x 120 mesh.



(a)

(b)

Figure 4.20 Flow past an idealised groyne for Re_I = 7.9 at t = 80 s computed using the curvilinear grid code (on a 360 x 60 Cartesian grid): (a) stream function contours; and (b) velocity vectors.

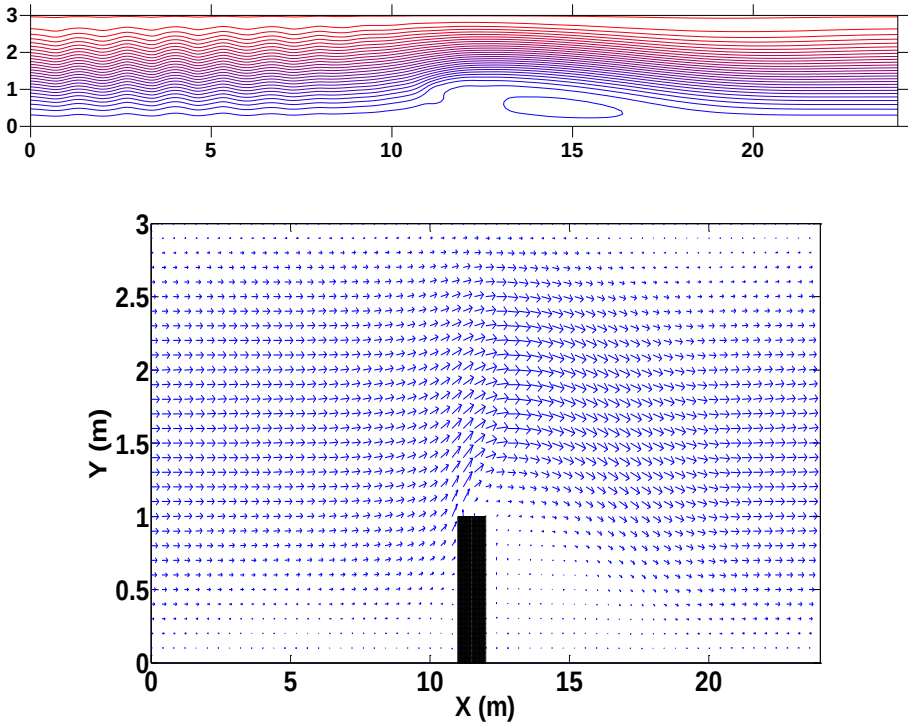


Figure 4.21 Flow past an idealised groyne for Re_I = 98 at t = 80 s computed using the curvilinear grid code (on a 360 x 60 Cartesian grid): (a) stream function contours; and (b) velocity vectors.

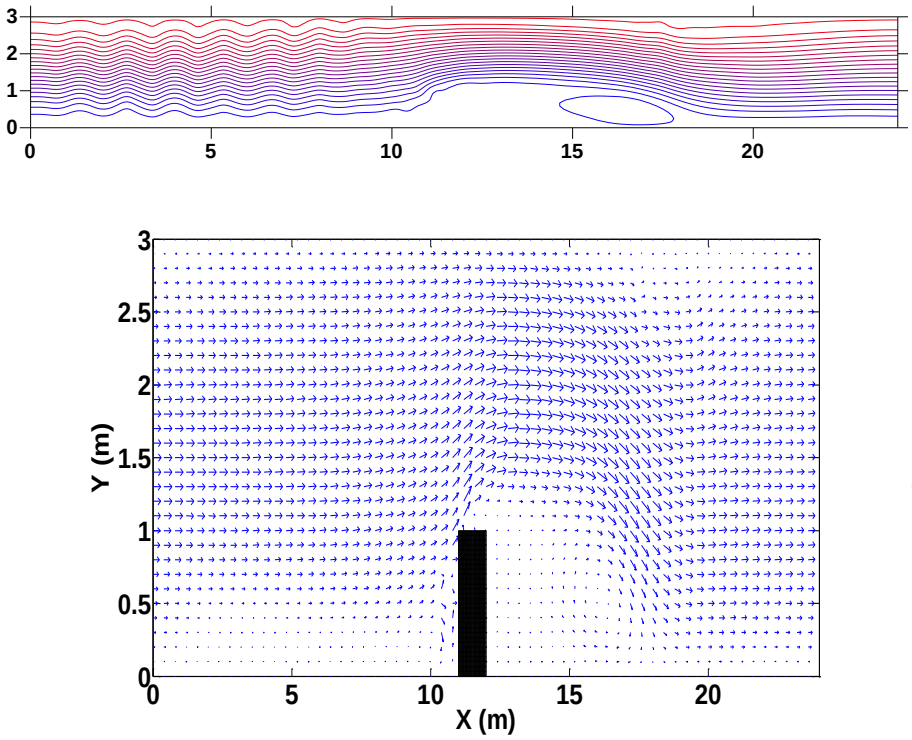


Figure 4.22 Flow past an idealised groyne for Re_I = 229 at t = 80 s computed using the curvilinear grid code (on a 360 x 60 Cartesian grid): (a) stream function contours; and (b) velocity vectors.

Chapter 5

Lagrangian Particle Tracking Model for Chaotic Advection

This chapter describes a Lagrangian particle tracking model that will later be coupled to the curvilinear shallow flow solver to investigate mixing processes in idealized river bends. The chapter commences by outlining the particle tracking methodology, and its verification for particle trajectories in the steady flow field of a free vortex and in the abruptly changing flow field described by a pair of blinking vortices. The numerical solver is partially validated by simulating flow in a pair of side-wall cavities against experimental data provided by Tuna and Rockwell (2014). Particle trajectories are simulated behind a pair of groynes in a rectangular channel, and the mixing processes evaluated. The mixing characteristics are determined using the mean particle separation distance, from which the Lyapunov exponent is evaluated.

5.1 Lagrangian particle tracking

In a fluid flow, the advection of a tracer is simply described by

$$u_p = \frac{\mathrm{d}x_p}{\mathrm{d}t} \quad \text{and} \quad v_p = \frac{\mathrm{d}y_p}{\mathrm{d}t} \qquad (5.1)$$

where (u_p, v_p) are the Cartesian components of velocity of a particle located at (x_p, y_p). For a passive tracer the particle velocity at a point in space and time is the same as the flow velocity there. Equation (5.1) is solved by numerical integration using a fourth-order Runge-Kutta time integration method as follows:

$$x_{p_{i,j}}^{k+1} = x_{p_{i,j}}^{k} + \frac{1}{6}\Delta t(k_{11} + 2\,k_{12} + 2\,k_{13} + k_{14}) \qquad (5.2a)$$

$$y_{p_{i,j}}^{k+1} = y_{p_{i,j}}^{k} + \frac{1}{6}\Delta t(k_{21} + 2\,k_{22} + 2\,k_{23} + k_{24}) \qquad (5.2b)$$

where:

$$k_{11} = \left[\frac{du_p}{dt}(t, u_p)\right] \qquad \text{and} \qquad k_{21} = \left[\frac{dv_p}{dt}(t, v_p)\right]$$

$$k_{12} = \left[\frac{du_p}{dt}\left(t + \frac{\Delta t}{2}, u_p + \frac{k_{11}}{2}\right)\right] \quad \text{and} \quad k_{22} = \left[\frac{dv_p}{dt}\left(t + \frac{\Delta t}{2}, v_p + \frac{k_{21}}{2}\right)\right]$$

$$k_{13} = \left[\frac{du_p}{dt}\left(t + \frac{\Delta t}{2}, u_p + \frac{k_{12}}{2}\right)\right] \quad \text{and} \quad k_{23} = \left[\frac{dv_p}{dt}\left(t + \frac{\Delta t}{2}, v_p + \frac{k_{22}}{2}\right)\right]$$

$$k_{14} = \left[\frac{du_p}{dt}(t + \Delta t, u_p + k_{13})\right] \quad \text{and} \quad k_{24} = \left[\frac{dv_p}{dt}(t + \Delta t, v_p + k_{23})\right]$$

In cases where the velocity field is known analytically as a continuous function, we simply set ($u_p(x_p,y_p,t)$, $v_p(x_p,y_p,t)$) = ($u(x_p,y_p,t)$, $v(x_p,y_p,t)$) where u and v are the flow velocity components. In cases where the velocity field is computed at discrete points on a grid, the particle velocity is calculated using linear interpolation method. Figure 5.1 illustrates the template used to determine the velocity $(u_p,\ v_p)$ for a particle at $(x_p,\ y_p)$. The interpolation process involves three stages. First, the cell containing the particle is identified so $i = i_p$ and $j = j_p$ when $x_i < x_p < x_{i+1}$ and $y_i < y_p < y_{i+1}$. We then determine (u_1, v_1) at the south side of the cell at a position aligned with $(x_p,\ y_{jp})$ as follows:

$$u_1 = u_{ip,jp} + \left(u_{ip+1,jp} - u_{ip,jp}\right)\frac{\left(x_p - x_{ip,jp}\right)}{\left(x_{ip+1,jp} - x_{ip,jp}\right)} \qquad (5.3a)$$

$$v_1 = v_{ip,jp} + \left(v_{ip+1,jp} - v_{ip,jp}\right)\frac{\left(x_p - x_{ip,jp}\right)}{\left(x_{ip+1,jp} - x_{ip,jp}\right)} \qquad (5.3b)$$

And we also determine (u_2, v_2) at the north side of the cell at a position aligned with $(x_p,\ y_{jp+1})$ as follows:

$$u_2 = u_{ip,jp+1} + \left(u_{ip+1,jp+1} - u_{ip,jp+1}\right)\frac{\left(x_p - x_{ip,jp+1}\right)}{\left(x_{ip+1,jp+1} - x_{ip,jp+1}\right)} \qquad (5.4a)$$

$$v_2 = v_{ip,jp+1} + \left(v_{ip+1,jp+1} - v_{ip,jp+1}\right)\frac{\left(x_p - x_{ip,jp+1}\right)}{\left(x_{ip+1,jp+1} - x_{ip,jp+1}\right)} \qquad (5.4b)$$

The third step involves interpolation in the y-direction, giving (u_p, v_p) as:

$$u_p = u_1 + (u_2 - u_1)\frac{(y_p - y_{ip,jp})}{(y_{ip,jp+1} - y_{ip,jp})} \qquad (5.5a)$$

$$v_p = v_1 + (v_2 - v_1)\frac{(y_p - y_{ip,jp})}{(y_{ip,jp+1} - y_{ip,jp})} \qquad (5.5b)$$

A similar approach is undertaken to interpolation on the curvilinear grid.

5.2 Particle trajectory in the flow field of a free vortex

From potential theory, the stream function of a free vortex is defined as

$$\psi = -\frac{\Gamma}{2\pi}\ln r \qquad (5.6)$$

in which the radial distance, $r = \sqrt{(x^2 + y^2)}$, and Γ is the vortex strength. In the polar coordinate system, the radial and tangential velocity components are:

$$u_r = 0 \quad \text{and} \quad u_\theta = \frac{\Gamma}{2\pi r} \qquad (5.7)$$

The Cartesian and polar velocity vector components are related as follows.

$$u = u_r \cos\theta - u_\theta \sin\theta \qquad (5.8a)$$

and

$$v = u_r \sin\theta + u_\theta \cos\theta \qquad (5.8b)$$

where $\theta = \tan^{-1}\frac{x}{y}$. In short,

$$u = -\frac{\Gamma}{2\pi r}\sin\theta \quad \text{and} \quad v = \frac{\Gamma}{2\pi r}\cos\theta \qquad (5.9)$$

Figure 5.2 shows the velocity vectors and velocity profile based on equation (5.9) for $\Gamma = 2\pi$ that has been computed on 21 x 21 grids with spacing Δx = Δy = 0.5 m. Consider a particle that is initially at (x_p, y_p) = (1.23, 0.82), such that its radial distance

from the origin of the vortex is 1.4782760 m. Its initial velocity is (u_p, v_p) = (-0.01896, 0.02855) m/s. Figure 5.3 shows that the particle trajectory, computed using velocity components derived by linear interpolation using equations (5.3) to (5.5) from the discrete grid values, forms an almost closed circular loop. The time step is Δt = 0.1 s and the simulation time is t = 400 s, at which time the numerically predicted radial distance of the particle from the origin of the vortex is 1.4786591 m. This test confirms that the linear interpolation and particle tracking algorithms are working correctly for a steady flow problem.

5.3 Particle advection due to a blinking-vortex pair

To analyze two-dimensional particle advection in the abruptly changing flow field described by a pair of blinking vortices in a closed circular container, Aref (1984) assumed a dynamic Hamiltonian system. Aref derived an analytical solution for the integrated particle trajectory from the pure advection equations for a Hamiltonian system. Aref was then able to determine particle patch advection patterns with time, the non-linear dynamics through Poincaré maps, etc., and the trend towards chaotic advection as a vortex strength parameter was increased. Later, Khakhar and Ottino (1986) repeated much the same exercise for particle advection due to a pair of vortex stirrers in an otherwise infinite expanse of fluid. The present study effectively follows the approach taken by Khakhar and Ottino. The first vortex is centred at (x_1, y_1) = (-0.5, 0) and the second vortex at (x_2, y_2) = (0.5, 0). Both vortices are free, and their combined stream function through linear superposition is:

$$\psi = \psi_1 + \psi_2 = -\frac{\Gamma_1}{2\pi r_1}\ln r_1 - \frac{\Gamma_2}{2\pi r_2}\, ln\, r_2 \qquad (5.10)$$

where:

$$\Gamma_1 = \Gamma_2 = 2\pi$$

and

$$r_1 = \sqrt{((x - x_1)^2 + (y - y_1)^2)} \quad \text{and} \quad r_2 = \sqrt{((x - x_2)^2 + (y - y_2)^2)} \quad .$$

The radial and tangential velocity components are:

$$u_r = u_{r1} + u_{r2} = 0 \quad , \qquad u_\theta = u_{\theta 1} + u_{\theta 2} = \frac{\Gamma_1}{2\pi r_1} + \frac{\Gamma_2}{2\pi r_2}$$

By applying the velocity vector components in the Cartesian and polar systems, we have:

$$u = u_1 + u_2 = -\frac{\Gamma_1}{2\pi r_1}\sin\theta_1 \; - \frac{\Gamma_2}{2\pi r_2}\sin\theta_2$$

$$v = \; v_1 + v_2 = +\frac{\Gamma_1}{2\pi r_1}\cos\theta_1 \; + \frac{\Gamma_2}{2\pi r_2}\cos\theta_2$$

By definition,

$$\sin\theta_1 \; = \frac{y - y_1}{r_1} \quad \text{and} \quad \cos\theta_1 \; = \frac{x - x_1}{r_1}$$

$$\sin\theta_2 \; = \frac{y - y_2}{r_1} \quad \text{and} \quad \cos\theta_2 \; = \frac{x - x_2}{r_2}$$

Hence,

$$u = u_1 + u_2 = -\frac{\Gamma_1}{2\pi r_1}\frac{(y - y_1)}{r_1} \; - \frac{\Gamma_2}{2\pi r_2}\frac{(y - y_2)}{r_2} \qquad (5.11a)$$

$$v = \; v_1 + v_2 = \frac{\Gamma_1}{2\pi r_1}\frac{(x - x_1)}{r_1} + \frac{\Gamma_2}{2\pi r_2}\frac{(x - x_2)}{r_2} \qquad (5.11b)$$

The blinking vortex model operates by switching on the first vortex, with the second off, for half a cycle during $nT < t < nT+T/2$ where $n = 0, 1, 2$ .... and T is the prescribed cycle period. The flow field is abruptly switched at $t=nT+T/2$ so that the second vortex is turned on while the other one is off during the latter half of the cycle when nT +T/2 $< t < (n+1)T$. The process is repeated for $n = 0, 1, 2$, ... until the simulation is complete.

Figures 5.4a and 5.4b illustrate the velocity vector fields produced by a typical pair of blinking vortices during each half cycle.

Aref (1984) defines a dimensionless parameter μ as follows:

$$\mu = \frac{\Gamma T}{2\pi a^2} \qquad (5.12)$$

where T is the blinking period, Γ is the vortex strength, and a is the distance of the centre of the blinking vortex from the origin. By setting $\Gamma = 2\pi$ and $a = 1.0$, the dimensionless parameter μ is equal to the period T. The choice of value for μ has a profound effect on the long-term dynamic behaviour of particles.

Following Aref (1984), we consider blinking vortices located at $(x_1, y_1) = (-0.5,\ 0.0)$ and $(x_2, y_2) = (0.5,\ 0.0)$ and the trajectories of 15 particles whose initial positions are shown in Figure 5.5 such that $x_p = \pm 0.05, \pm 0.2, \pm 0.35$ and $y_p = 0.0$, and $y_p = 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8,\ 0.9$ and $x_p = 0.0$. Figure 5.6 shows stroboscopic maps obtained analytically by Aref (1984) as superimposed snapshots of the particle positions taken exactly one cycle apart for different values of μ for time up to $500T$. For $\mu = 0.05$, the motion of the particles is generally periodic and observes closed orbits. For $\mu \geq 0.1$, chaotic regions develop close to the blinking vortices. As μ increases further, the chaotic region grows until it fills much of the computational domain.

Figure 5.7 presents the stroboscopic maps obtained for the same cases as considered by Aref, but using the present Lagrangian particle tracking model using analytical values of the flow velocity components determined from Equations (5.11a) and (5.11b). Figure 5.8 shows the corresponding stroboscopic maps obtained using the present Lagrangian particle tracking model with values of flow velocity determined by interpolation from a 21 by 21 grid in a square domain such that $x = \in (-1.5, 1.5)$ and $y = \in (-1.5, 1.5)$. The results are in close agreement with those in Figure 5.7, indicating that the interpolation scheme is working satisfactorily. The results are also in reasonable qualitative agreement with those obtained by Aref, though it should be noted that his model was developed for chaotic advection driven by stirrers in a circular

container, whereas the present model is for chaotic advection driven by stirrers in an infinite domain.

Next, consider the trajectories of an initially square array of 10000 particles located at the origin of the domain midway between the centres of the two vortices. The particles are tracked until $t = 12T$. Figure 5.9 shows the evolution of the particle patterns with time for $\mu = 1$ obtained by Aref (1984). Figures 5.10 and 5.11 show the corresponding results obtained using the present particle-tracking model with the velocity field obtained analytically (Figure 5.10) and numerically (Figure 5.11). There is good agreement between the three sets of results, again verifying the present numerical model. As time progresses, it can be seen that the square array initially becomes distended to form whorls and tendrils, after which chaotic advection causes the particle distribution to spread out and become homogeneous.

Figures 5.12 to 5.14 show the particle positions after different numbers of blinking vortex cycles where an initial square array of 6400 particles is divided into four quadrants, each of different colour for $\mu = 0.05$, 0.35, and 1. The plots show that mixing is more rapid and widespread the higher the value of μ considered. Transition to chaos is evident in regions near the stirrers, with whorls and tendrils evident at the periphery.

5.4 Particle advection in a pair of adjacent side-wall cavities

To validate further the numerical model, we consider alternative numerical model and laboratory data obtained by Tuna and Rockwell (2014) who determined the vorticity distribution within oscillatory flow past a side-wall expansion containing an idealized groyne in a rectangular open channel. In the physical model, the flow enters from left side of channel. Figure 5.15 shows a plan view of the model used by Tuna and Rockwell. The main channel is 700 mm long and 475 mm wide. After an initial reach of 200 mm, the channel undergoes a side-wall expansion of length 300 mm and width 225 mm, and contains a groyne of 10 mm length and 225 mm width that is oriented perpendicular to the channel wall. After the side-wall expansion, the channel again becomes 475 mm wide. The mean water depth is 38mm. Two cases were considered by Tuna and Rockwell involving respectively an inlet velocity of U_1 = 273 mm/s and

U_2 =475mm/s. Figure 5.15c shows the streamlines in a pair of adjacent side-wall cavities which is obtained from Tuna and Rockwell (2014).

The present Cartesian shallow flow solver has been implemented to reproduce the flow conditions in a qualitatively similar configuration to that considered by Tuna and Rockwell. A uniform grid is placed over the whole domain (measuring 700 mm by 700 mm) covering the open channel, with patches of bed roughness used to model the non-fluid parts of the domain, and the bed roughness coefficient set to *cf* = 100; the bed roughness elsewhere is zero. Also, similar to Barber's study (1990) the eddy viscosity is computed through using depth-averaged eddy viscosity formula:

$$v = 5.9h\frac{\sqrt{g(U^2+V^2)}}{C} \qquad (5.13)$$

Where, $C = \frac{h^{\frac{1}{6}}}{n} = \frac{0.038^{1/6}}{0.03} = 19.32$ mm^{½}/s. Considering equation (5.13), the eddy viscosities are 9.992 mm^2/s and 17.264 mm^2/s corresponding to inlet velocities of U_1 = 273 mm/s and U_2 = 475 mm/s. The total simulation time is 20 s. Figure 5.16 shows the velocity vectors and stream function contours obtained for an inlet velocity of 273 mm/s using two grids, one 140 x 140 with Δt = 0.01 s, the other 280 x 280 with Δt = 0.01 s; the results are in satisfactory agreement, implying grid convergence. Figure 5.17 shows the corresponding results for an inlet velocity of 475 mm/s and Δt = 0.001 s; again grid convergence appears to have been achieved. Both Figures 5.16 and 5.17 include plots showing the depth profile across the middle of the cavities (from west to east), showing that there is a small but progressive downward sloping gradient in depth across the pair of cavities. The results are qualitatively similar to those obtained by Tuna and Rockwell model (see Figure 5.15c).

Particle mixing takes place inside the cavities between the two side-wall expansions that are divided by the groyne, owing to the presence of two recirculation zones, one in each cavity. To visualize particle mixing, the space within each cavity is filled with four layers of particles, the layers containing red, blue, purple and green particles, respectively. By carrying out Lagrangian particle tracking on a 280 x 280 flow grid, it is possible to examine mixing promoted by the two recirculation zones. Here, the time step used by the Lagrangian particle tracker is 0.5 s. Figure 5.18 shows

snapshots of the particle positions at different times after their release, for a case where the inlet velocity is 273 mm/s. At first, the northern green and purple layers develop waves which then curl over and form whorls. The blue layer gradually gets pulled upwards too. As can be seen in Figure 5.19, similar results are obtained for the higher inlet velocity of 475 mm/s, though the roll up of the northern layers takes place more quickly.

5.4.1 Mean particle separation and Lyaponov exponent

An appropriate technique is needed in order to analyze the degree of mixing that takes place with time. With this in mind, many researchers have applied the Lyaponov exponent as a measure of mixing, after computing the mean particle separation distance of an initial array of particles as it evolves in time (Khakhar and Ottino, 1986; Liang *et al*. 2006). In the present work, particles that have north and south adjacent particles are first selected. Then the mean particle separation is estimated, and plotted against time, from which the Lyaponov exponent is obtained as:

$$\sigma = \lim_{\substack{t\to\infty \\ d(0)\to 0}} \left(\frac{1}{t}\right) \ln \frac{d(t)}{d(0)} \qquad (5.14)$$

where σ is the Lyapunov exponent, d is separation distance, and t is time. Figure 5.20 indicates the mean particle separation as a function of time for the case considered in Figure 5.19. It is clear that the purple and green layers achieve the highest rates of dispersion.

5.5 Mixing of particles in a rectangular channel with groynes

The coupled shallow water and particle-tracking model is now used to simulate particle advection in a rectangular open channel containing a pair of groynes. Three cases are considered. In Case *R*1, the rectangular channel has plan dimensions 36 m x 3 m, the groyne is modelled as a patch of increased roughness of dimensions 1 x 1 m. the value of groyne roughness is determined by using trial-error method to find the best flow separation and recirculation zone in the vicinity of groyne. Here, *cf* = 50, the inlet

velocity is set to 0.5 m/s, Reynolds number is 100 water depth is 1 m, and the eddy viscosity is determined by using Reynolds formula, $\upsilon = \frac{U_I L_{groyne}}{Re_I} = 0.005$ m^2/s.

The solution was computed on a 360 x 30 grid, with time step of 0.05 s, and total simulation time of 40 s. In Case *R*2, the rectangular channel has dimensions 360 m x 30 m, the groyne comprises a patch of dimensions 10 x 10 m and *cf* = 5. Again the inlet velocity is set to 0.5 m/s, Reynolds number is 100, water depth is 1 m, and the eddy viscosity is set to 0.05 m^2/s. The time step is 0.5 s, and the total simulation time is 400 s. In Case *R*3, the channel dimensions are 3600 m x 300 m, and the groyne is modelled as a patch of increased roughness of dimensions 100 x 100 m and *cf* = 0.5. The inlet velocity is 0.5 m/s, the water depth is 1 m, the eddy viscosity is 0.5 m^2/s, the time step is 5 s, and the (steady state) simulation time is 4000 s. Figures 5.21 and 5.22 show the velocity vectors and stream function contours at steady state for cases *R*1 to *R*3 obtained using the shallow flow solver. To study particle mixing in the vicinity of the groynes, the particle tracking model is used to simulate the trajectories taken by an initially rectangular array of 8000 particles with five colour bands released upstream of the first groyne (for flow case *R*3). Figure 5.23 shows the advected distributions of the particles at intervals of 100 s after the initial release until a simulation time of 1000 s is reached. The red particles are become stretched along a tendril, with the blue, magenta, green, and yellow particles more spread outs the position of particles from *t* = 0.0 s to *t* = 1000.0 s.

Figure 5.24 shows the rectangular channel containing two groynes, with particles initially filling the domain and also it depicts snapshots of particle distributions obtained at different times, until *t* = 1000 s. Figure 5.25 shows the temporal evolution of the mean particle separation distance for particles of the given colours. The yellow particles become most separated over the long term as they are advected by the main current in the channel. The red particles are initially the least separated because many are temporarily trapped in the recirculation zones.

5.6 Conclusion

This chapter has explained the principles behind the Lagrangian particle tracking model, which has been constructed as a numerical algorithm by integrating

forward in time the advection equations using a 4^{th}-order Runge-Kutta scheme. The Lagrangian particle tracking scheme has been verified successfully for the circular trajectory of a particle in the flow field of a free vortex, and for the progression to chaos in the alternating flow field of a pair of blinking vortices. Excellent agreement is achieved with the analytical model of Aref (1984), and the tests have shown that the present bilinear interpolation scheme works properly at converting the discretized flow field into an estimated continuous field. The model has been partly verified against laboratory data for open channel flow past a pair of adjacent side-wall cavities by Tuna and Rockwell (2014). Demonstration cases of particle tracking and mixing in an open channel containing two groynes have been investigated. The next chapter will apply the combined shallow water and Lagrangian particle tracking model to study advection and mixing processes in an idealized river bend similar to found along the Danube River, Hungary.

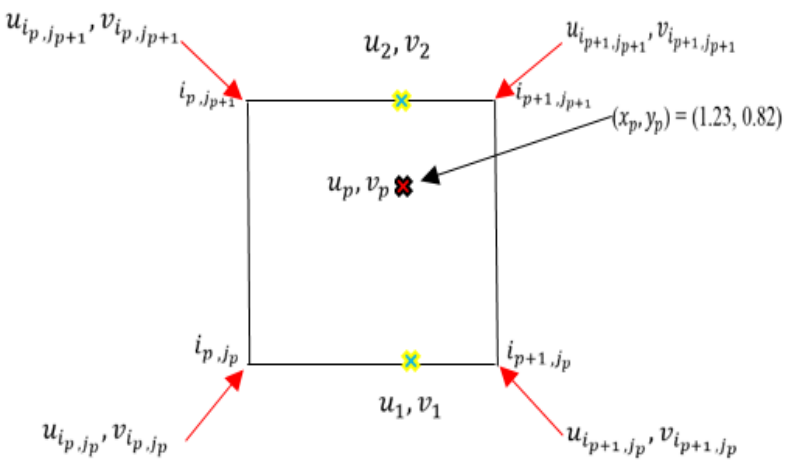
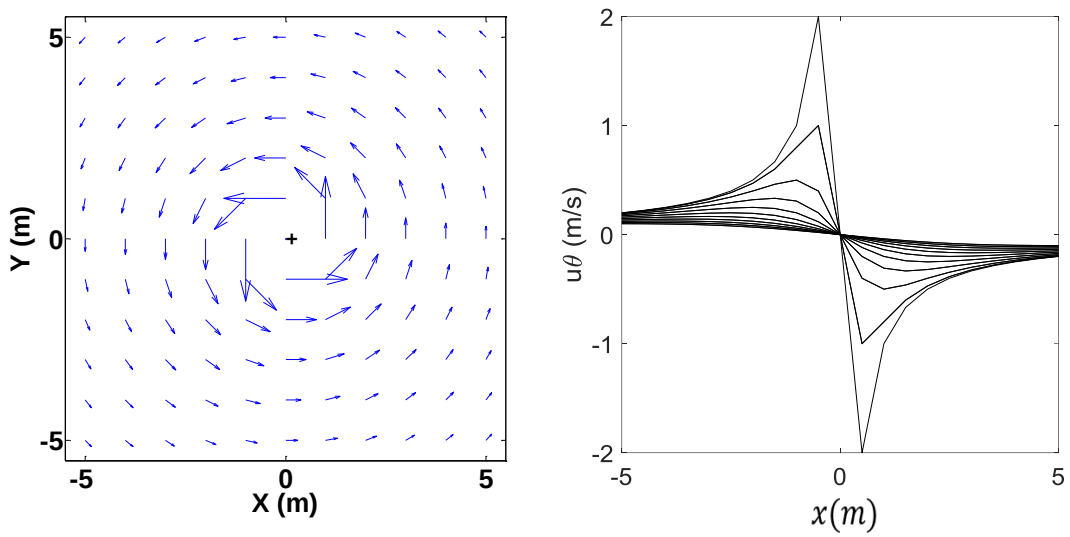


Figure 5.1 Linear interpolation to estimate velocity components at the point denoted (x_p, y_p) in the cell identified such that its southwest corner is at i_p, j_p.



(a) (b)

Figure 5.2 Flow induced by a free vortex: (a) velocity vectors, and (b) velocity profile along the $y = 0$ axis.

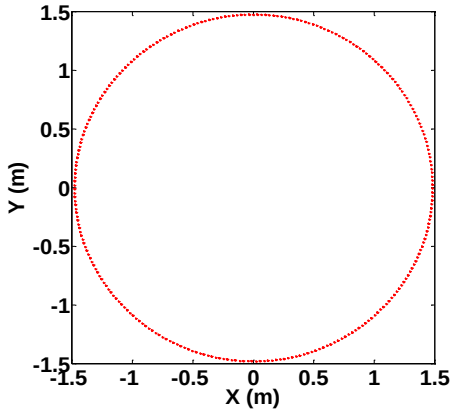
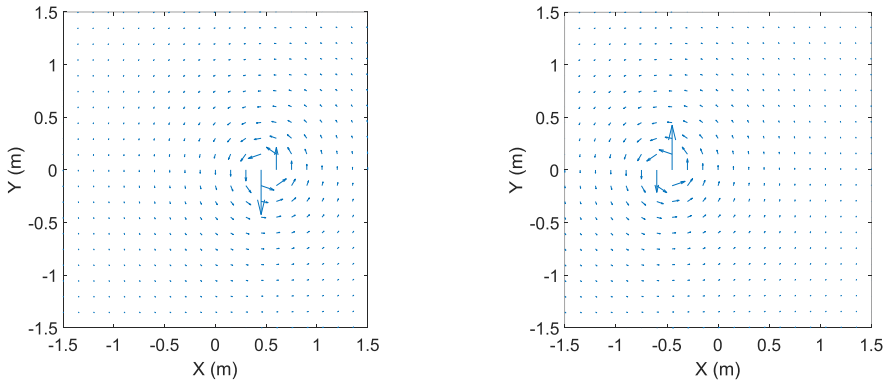


Figure 5.3 Trajectory of a particle initially at (x_p, y_p) = (1.23, 0.82) in the flow field of a free vortex of strength $\Gamma = 2\pi$.



(a) (b)

Figure 5.4 Velocity field for blinking vortices: at (a) $nT < t < nT+T/2$, and (b) nT +T/2 < t < $(n+1)T$ where n = 0, 1, 2, 3, 4.

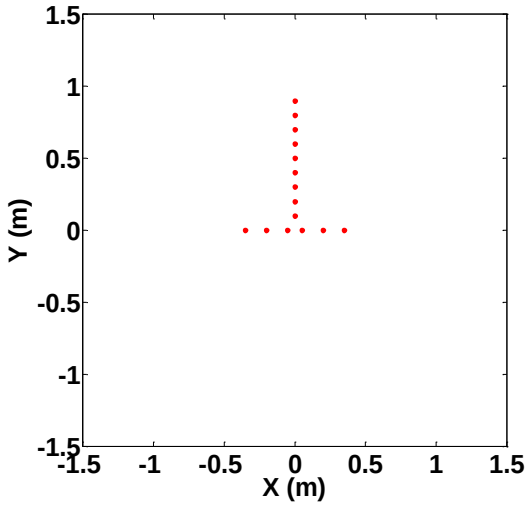


Figure 5.5 Initial position of 15 particles for blinking vortex cases

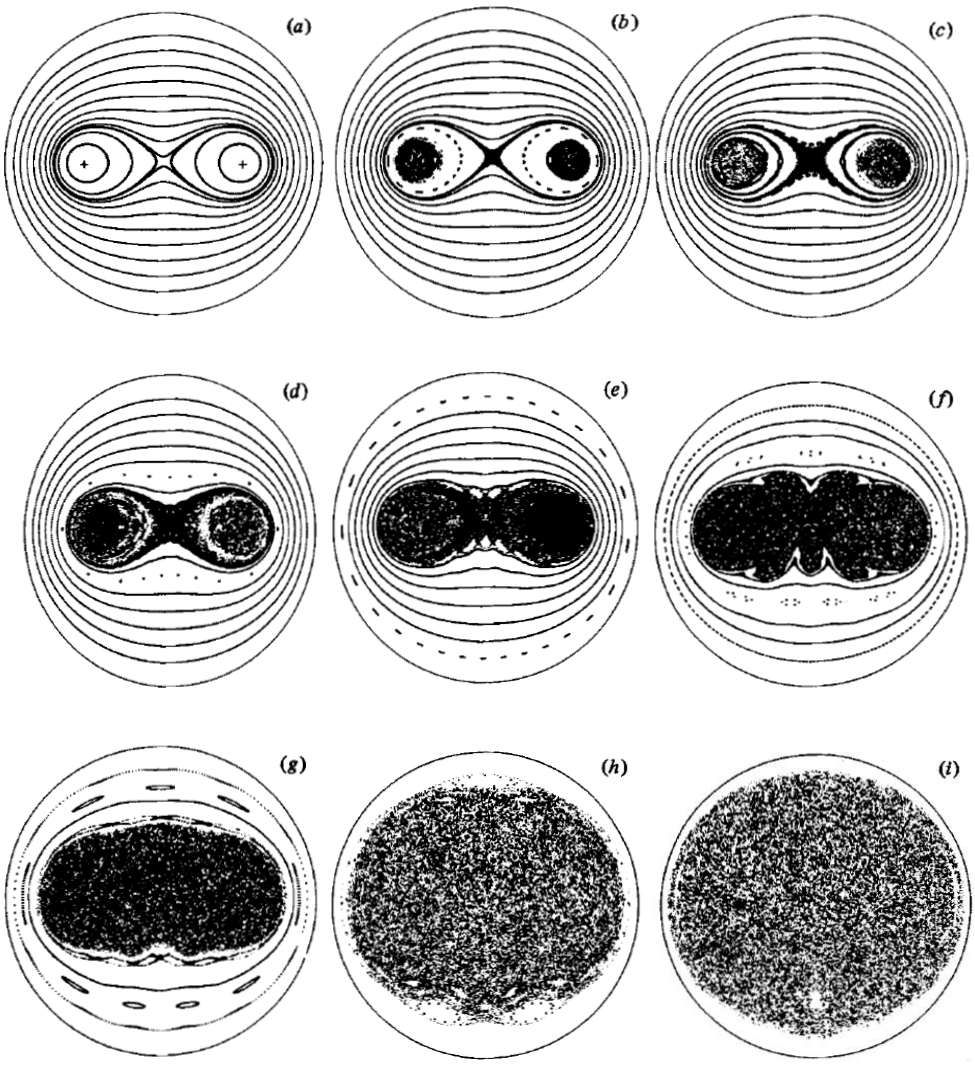
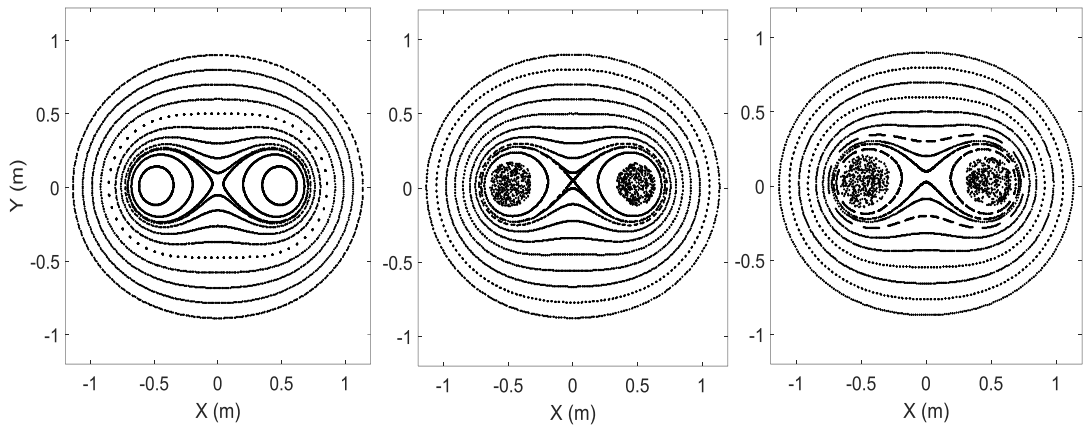
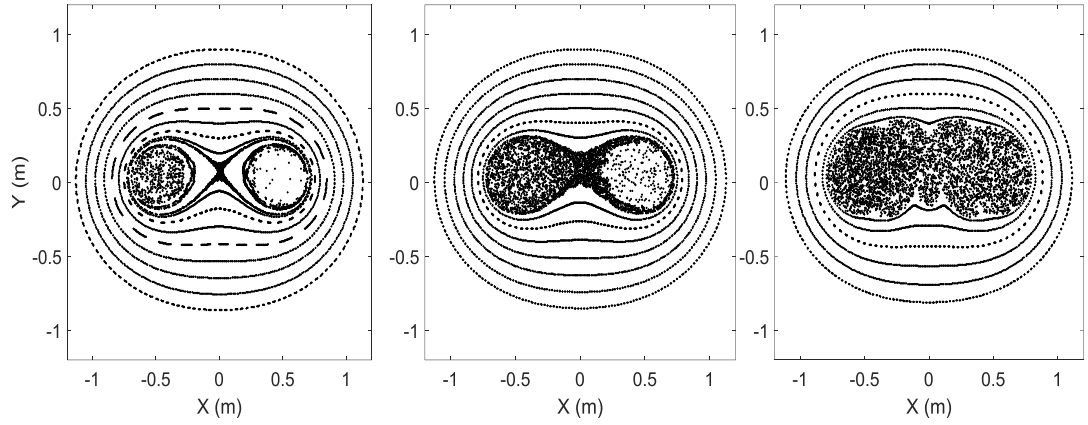


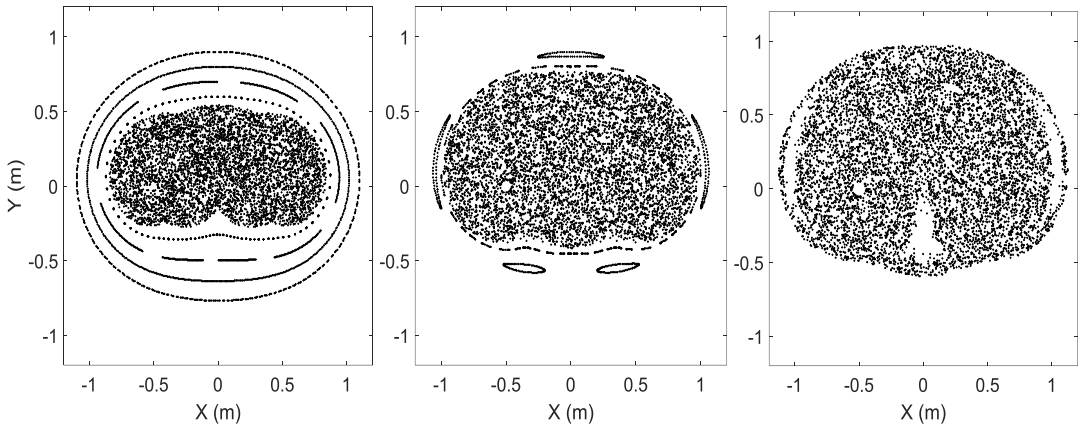
Figure 5.6 Analytically predicted stroboscopic maps for blinking vortex cases (Aref, 1984): (a) $\mu = 0.05$, (b) $\mu = 0.1$, (c) $\mu = 0.125$, (d) $\mu = 0.15$, (e) $\mu = 0.2$, (f) $\mu = 0.35$, (g) $\mu = 0.5$, (h) $\mu = 1.0$, and (i) $\mu = 1.5$.



(a) (b) (c)



(d) (e) (f)



(g) (h) (i)

Figure 5.7 Numerically predicted stroboscopic maps with the velocity field obtained analytically for blinking vortex cases: (a) $\mu = 0.05$, (b) $\mu = 0.1$, (c) $\mu = 0.125$, (d) $\mu = 0.15$, (e) $\mu = 0.2$, (f) $\mu = 0.35$, (g) $\mu = 0.5$, (h) $\mu = 1.0$, and (i) $\mu = 1.5$.

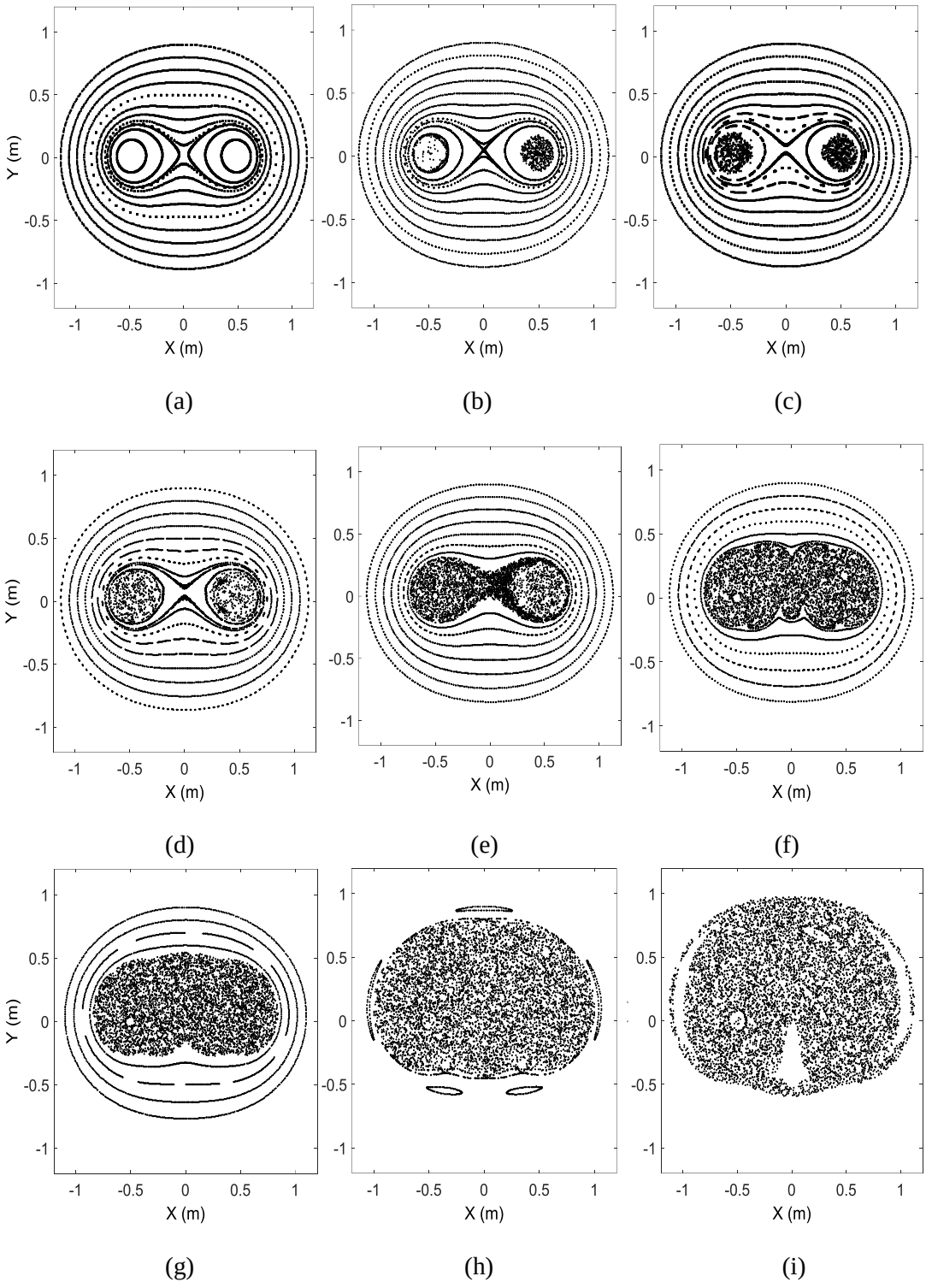


Figure 5.8 Numerically predicted stroboscopic maps with the velocity field obtained by bilinear interpolation for blinking vortex cases: (a) $\mu = 0.05$, (b) $\mu = 0.1$, (c) $\mu = 0.125$, (d) $\mu = 0.15$, (e) $\mu = 0.2$, (f) $\mu = 0.35$, (g) $\mu = 0.5$, (h) $\mu = 1.0$, and (i) $\mu = 1.5$.

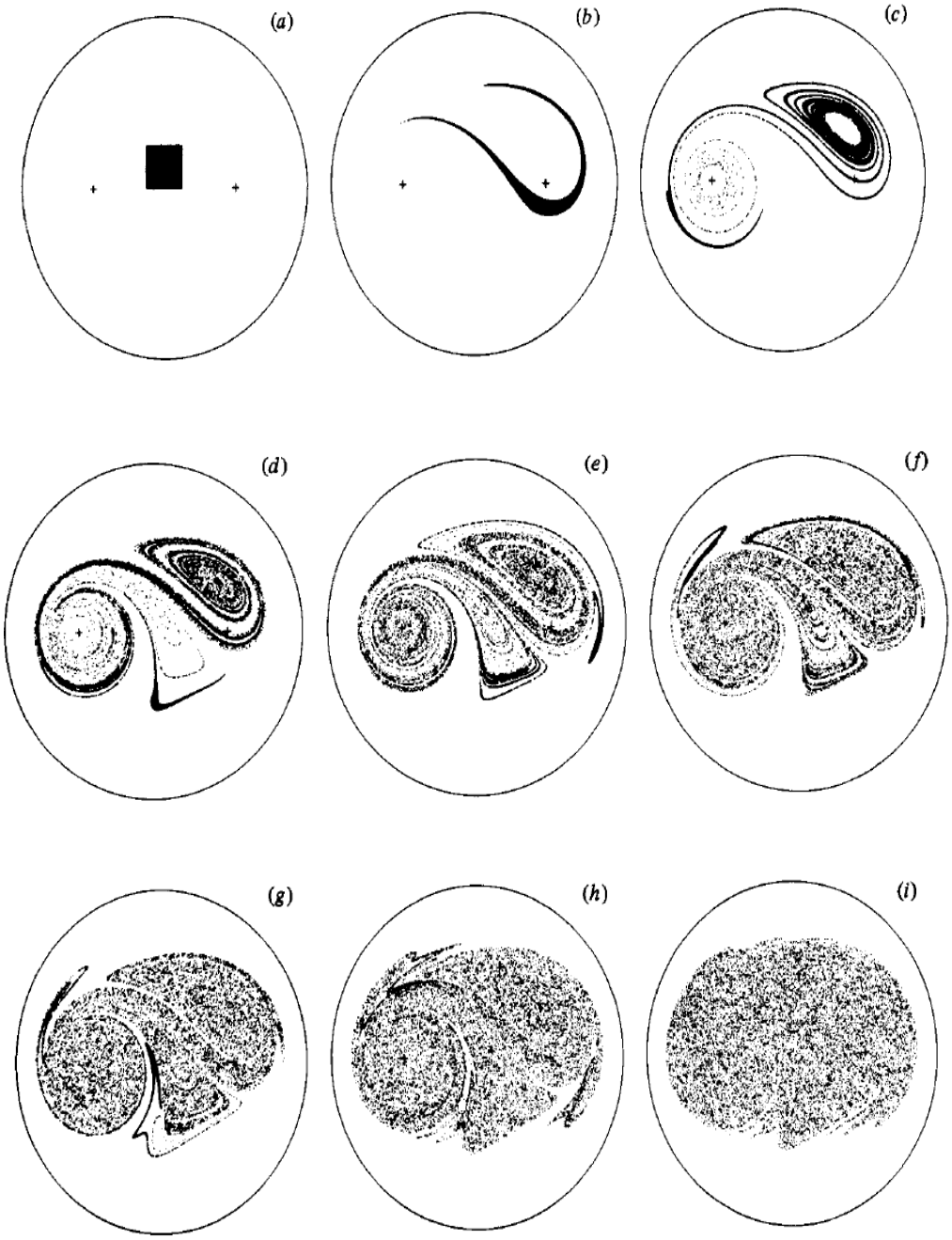
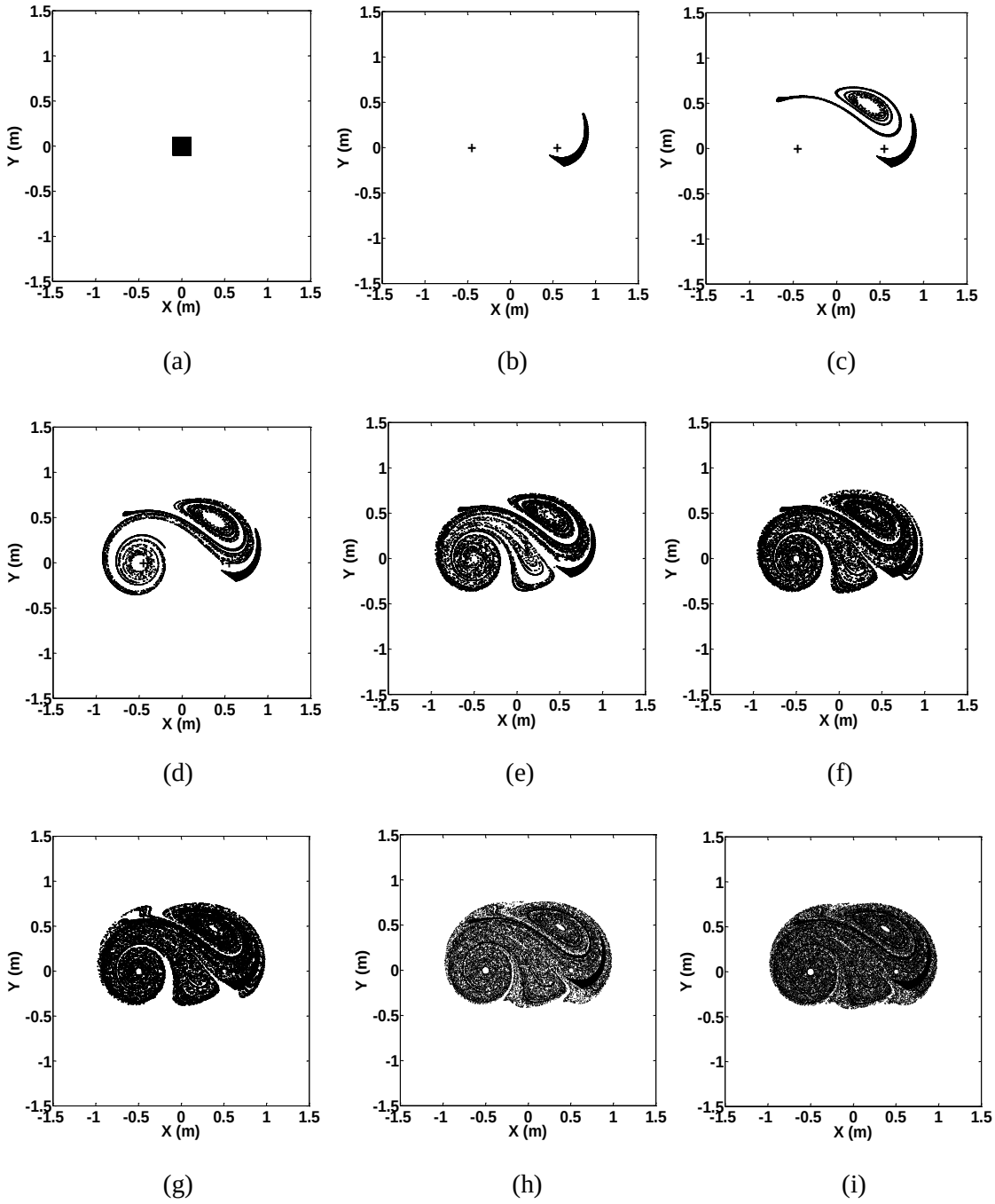


Figure 5.9 Analytical trajectories of 10,000 particles in an initially square array for blinking vortex cases (Aref, 1984): (a) $t = 0.0T$, (b) $t = 1.0T$, (c) $t = 2.0T$, (d) $t = 3.0T$, (e) $t = 4.0T$, (f) $t = 5.0T$, (g) $t = 6.0T$, (h) $t = 9.0T$, and (i) $t = 12.0T$.



(a) (b) (c)

(d) (e) (f)

(g) (h) (i)

Figure 5.10 Numerically predicted trajectories of 10,000 particles in an initially square array, with the velocity field obtained analytically for blinking vortex cases: (a) $t = 0.0T$, (b) $t = 1.0T$, (c) $t = 2.0T$, (d) $t = 3.0T$, (e) $t = 4.0T$, (f) $t = 5.0T$, (g) $t = 6.0T$, (h) $t = 9.0T$, and (i) $t = 12.0T$.

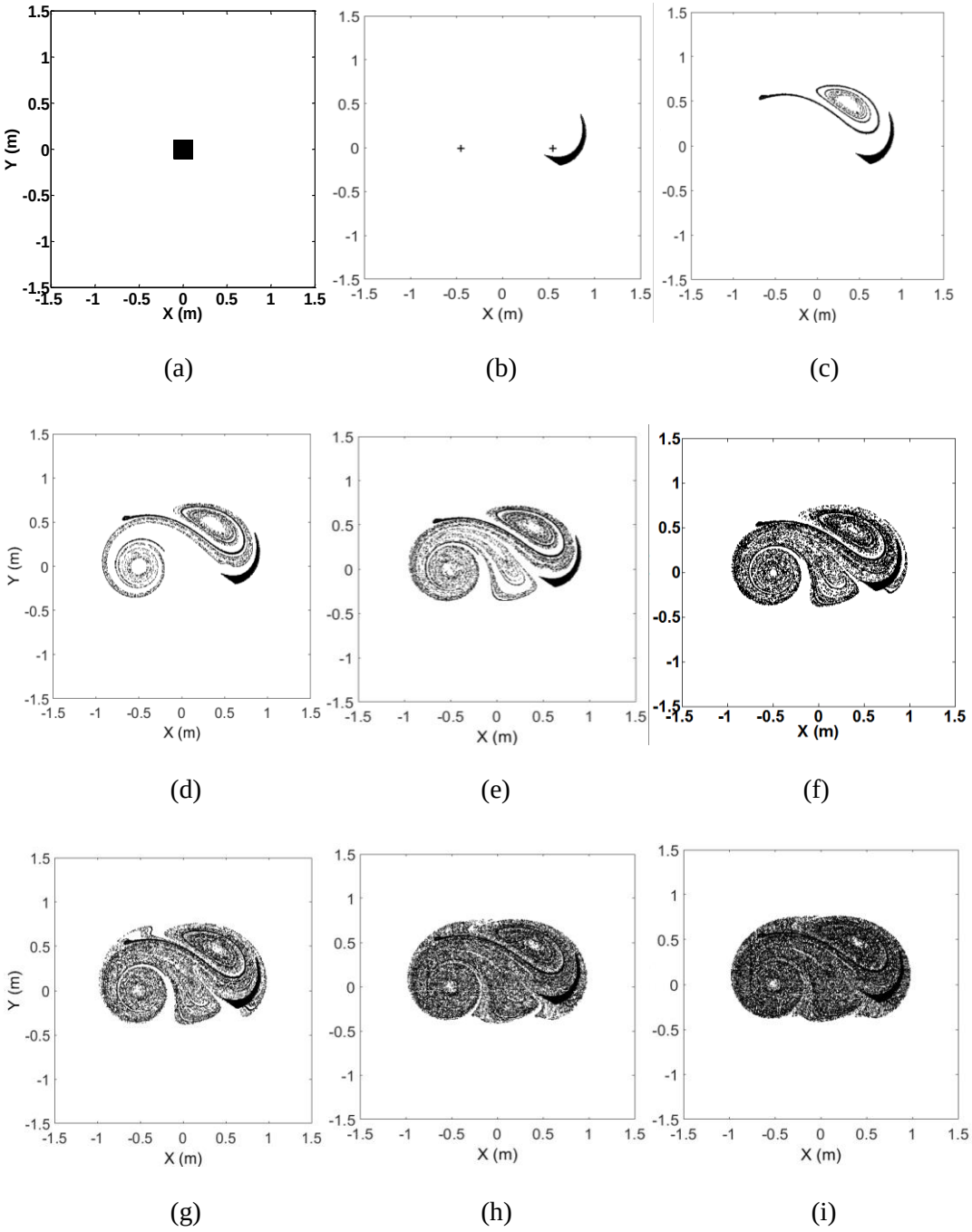


Figure 5.11 Numerically predicted trajectories of 10,000 particles in an initially square array, with the velocity field obtained using bilinear interpolation for blinking vortex cases: (a) $t = 0.0T$, (b) $t = 1.0T$, (c) $t = 2.0T$, (d) $t = 3.0T$, (e) $t = 4.0T$, (f) $t = 5.0T$, (g) $t = 6.0T$, (h) $t = 9.0T$, and (i) $t = 12.0T$.

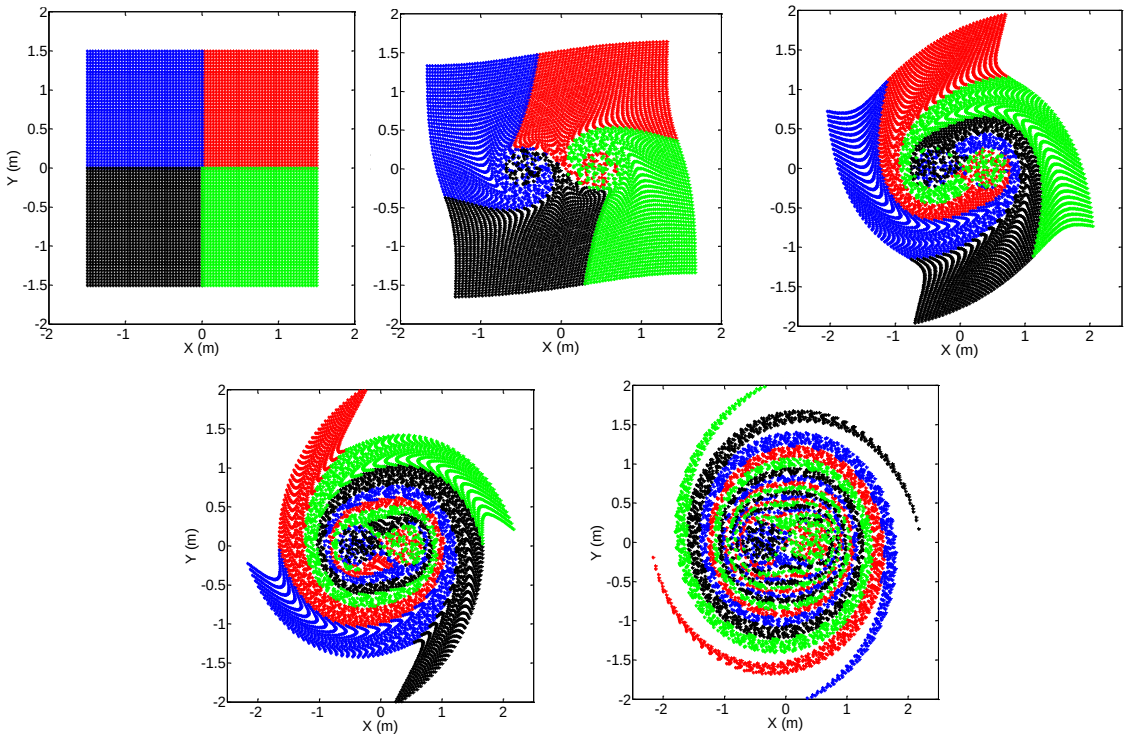


Figure 5.12 Numerically predicted particle positions with the velocity field obtained analytically for blinking vortex problem with $\mu = 0.05$: (a) $t = 0.0T$, (b) $t = 10.0T$, (c) $t = 40.0T$, (d) $t = 80.0T$, and (e) $t = 200.0T$.

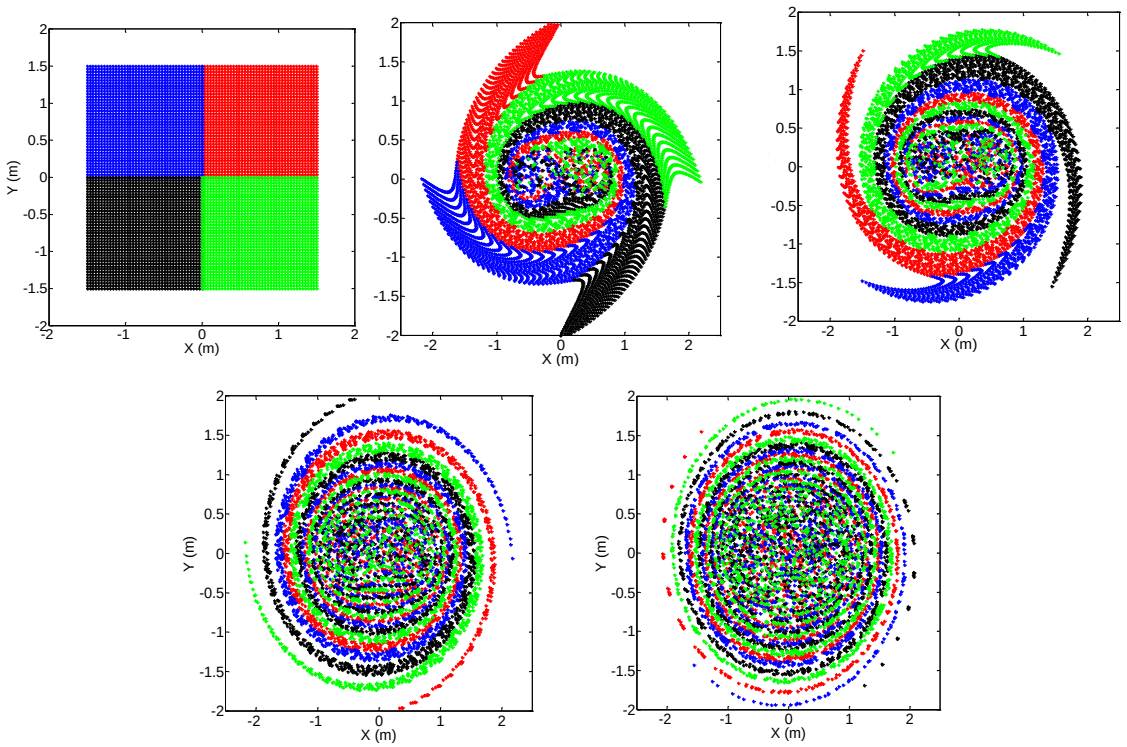


Figure 5.13 Numerically predicted particle positions with the velocity field obtained analytically for blinking vortex problem with $\mu = 0.35$: (a) $t = 0.0T$, (b) $t = 10.0T$, (c) $t = 20.0T$, (d) $t = 50.0T$, and (e) $t = 100.0T$.

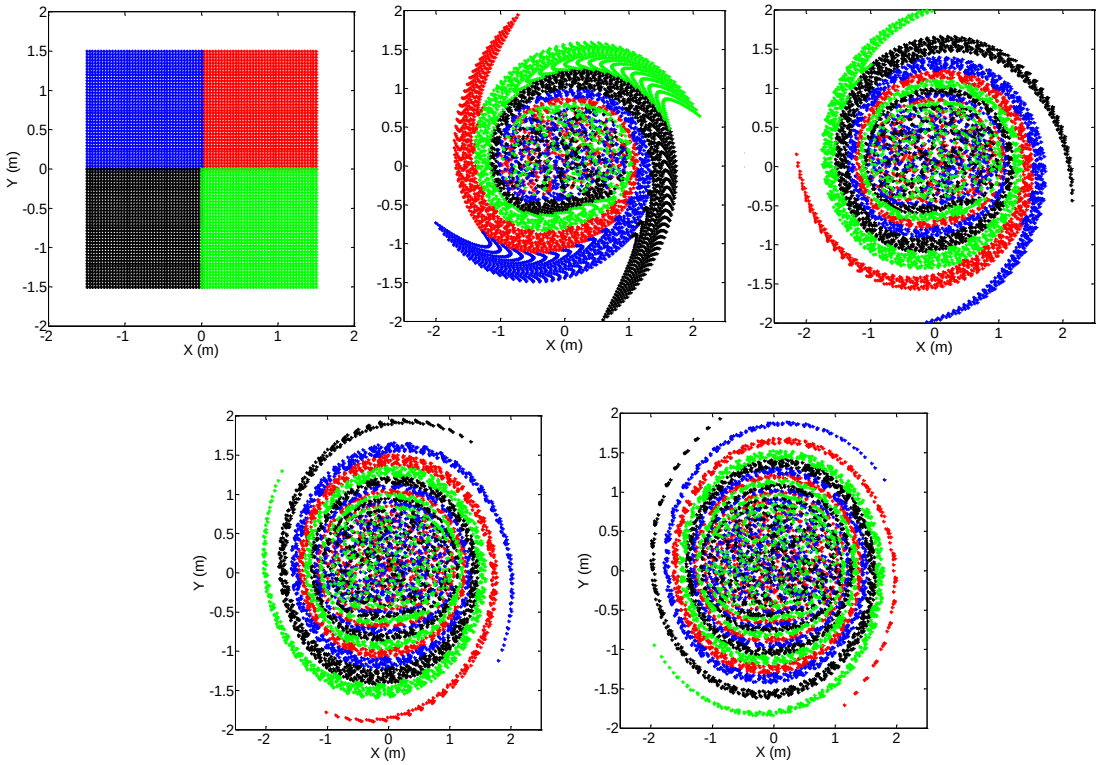
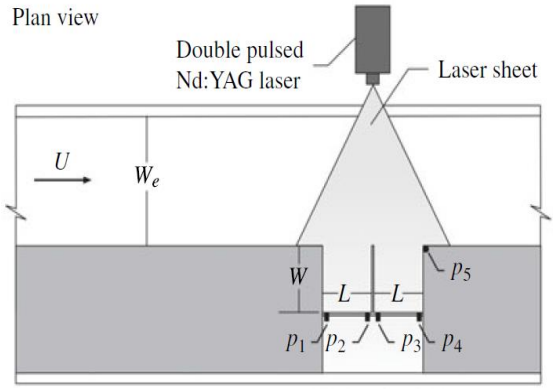
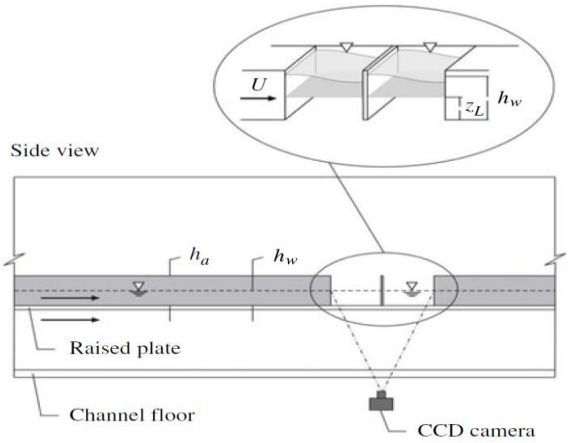


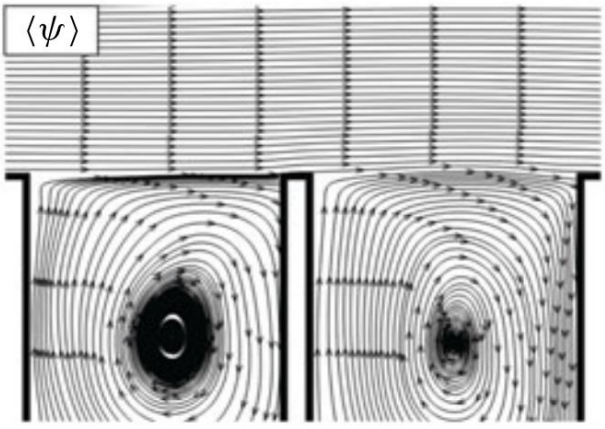
Figure 5.14 Numerically predicted particle positions with the velocity field obtained analytically for blinking vortex problem with $\mu = 1$: (a) $t = 0.0T$, (b) $t = 5.0T$, (c) $t = 10.0T$, (d) $t = 20.0T$, and (e) $t = 30.0T$.



(a)



(b)



(c)

Figure 5.15 Tuna and Rockwell Model for open channel flow past a pair of adjacent side-wall cavities: (a) Plan view, (b) side view of the laboratory side-wall cavities, and (c) Streamlines, Obtained from Tuna and Rockwell (2014).

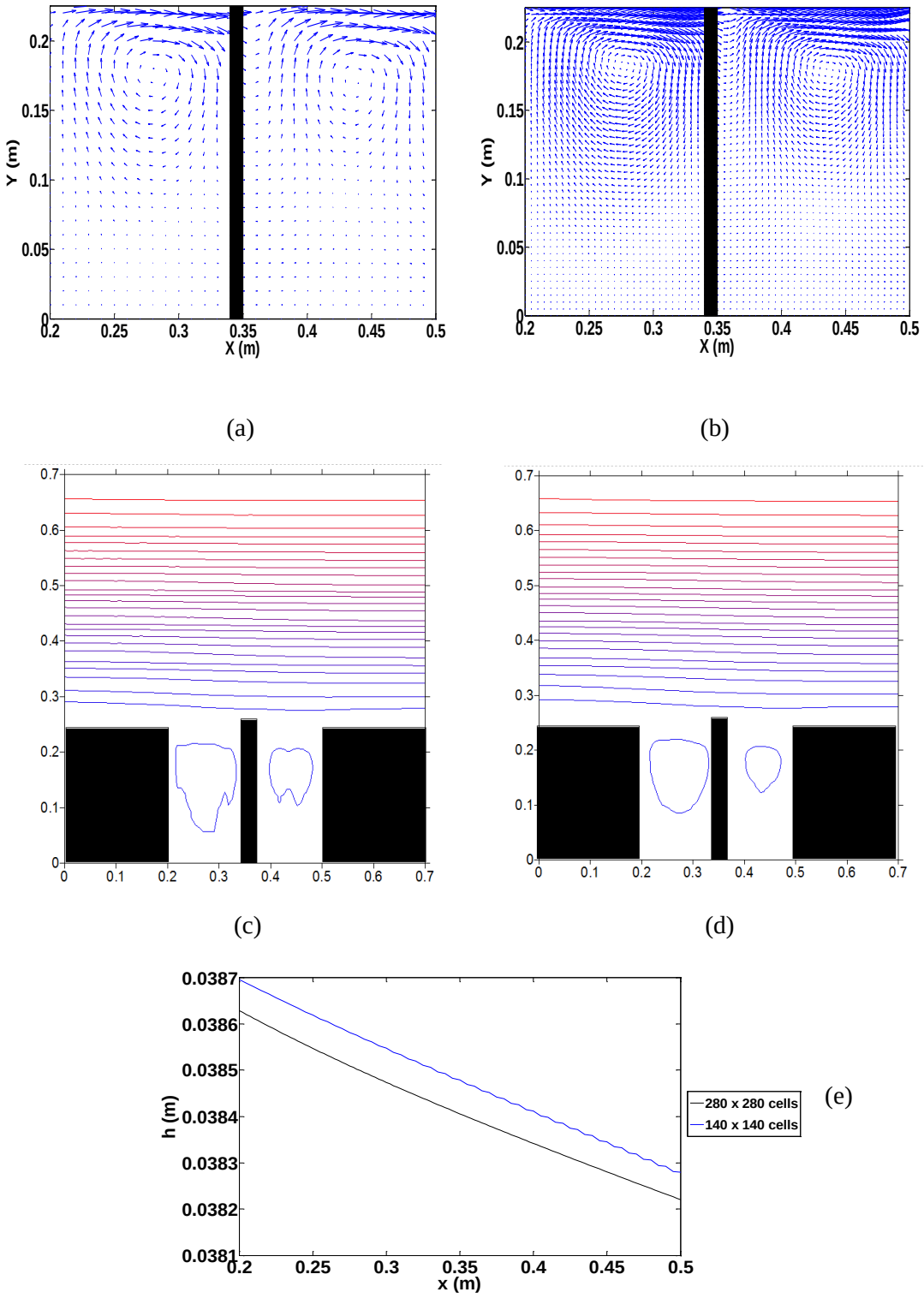


Figure 5.16 Open channel flow past a pair of adjacent side-wall cavities for U_{i1} = 273 mm/s at t = 20 s computed using the Cartesian grid code. Grid convergence test results showing: (a) velocity vectors on 140 x 140 mesh, (b) velocity vectors on 280 x 280 mesh, (c) stream function contours on 140x140 mesh, (d) stream function contours on 280 x 280 mesh; and (e) free surface elevation profiles along middle of cavities for 140 x 140 and 280 x 280 mesh.

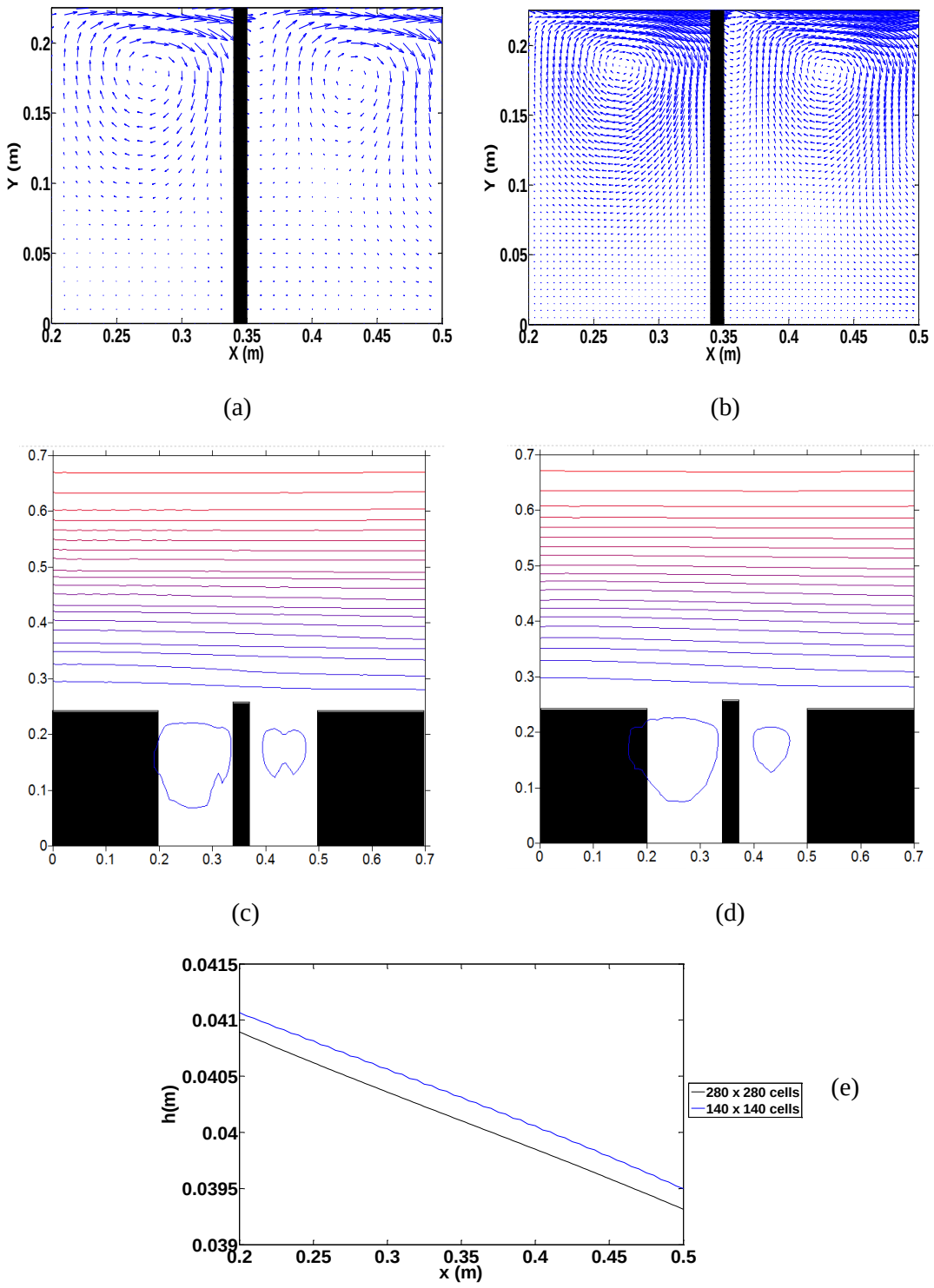


Figure 5.17 Open channel flow past a pair of adjacent side-wall cavities for U_{i2} = 475 mm/s at t = 20 s computed using the Cartesian grid code. Grid convergence test results showing: (a) velocity vectors on 140 x 140 mesh, (b) velocity vectors on 280 x 280 mesh, (c) stream function contours on 140x140 mesh, (d) stream function contours on 280 x 280 mesh; and (e) free surface elevation profiles along middle of cavities for 140 x 140 and 280 x 280 mesh.

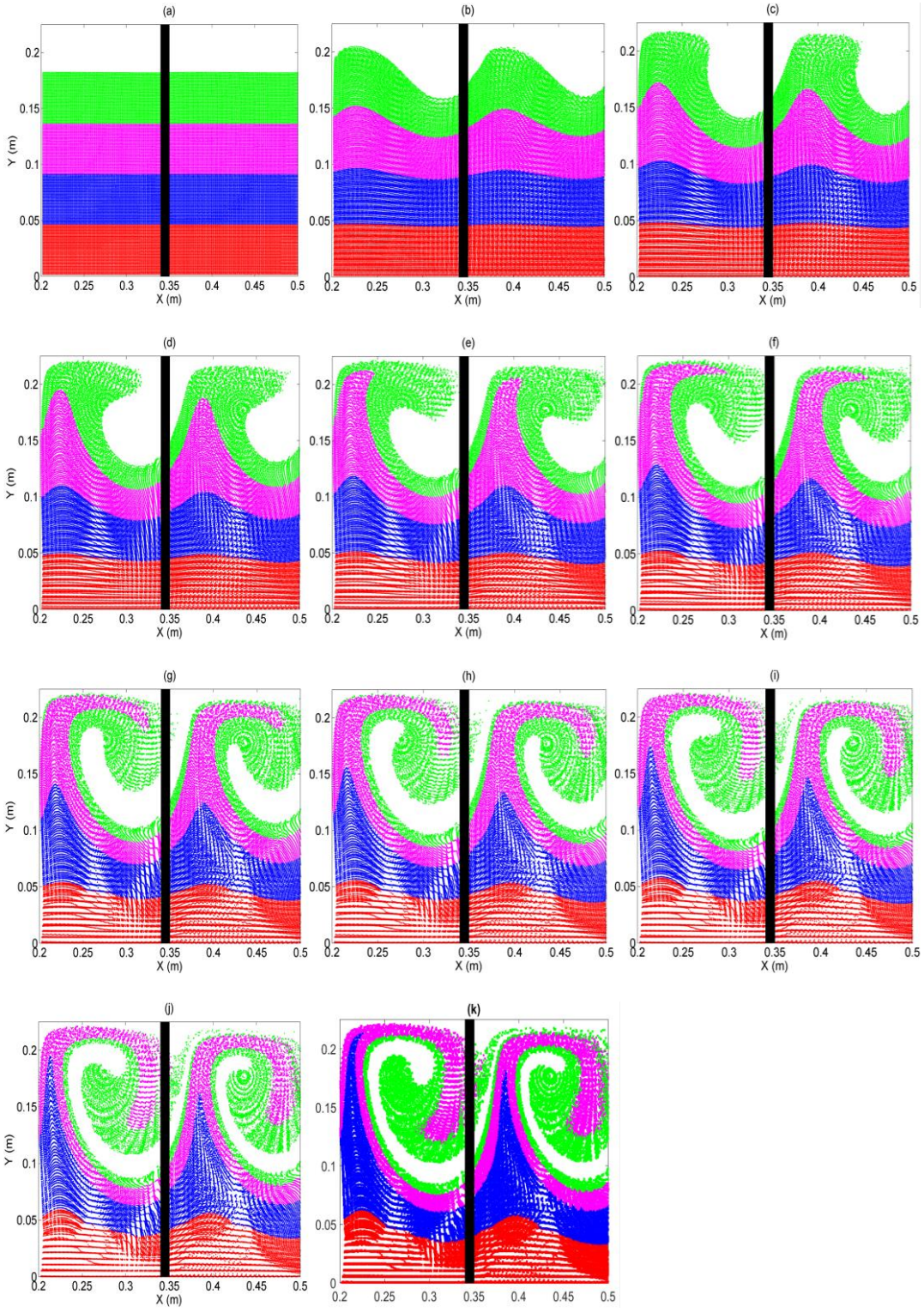


Figure 5.18 Particle advection in a pair of adjacent side-wall cavities for U_{i1} = 273 mm/s: (a) $t = 0.0s$, (b) $t = 5s$, (c) $t = 10s$, (d) $t = 15s$, (e) $t = 20s$, (f) $t = 25s$, (g) $t = 30.0s$, (h) $t = 35s$, (i) $t = 40s$, (j) $t = 45s$, and (k) $t = 50s$.

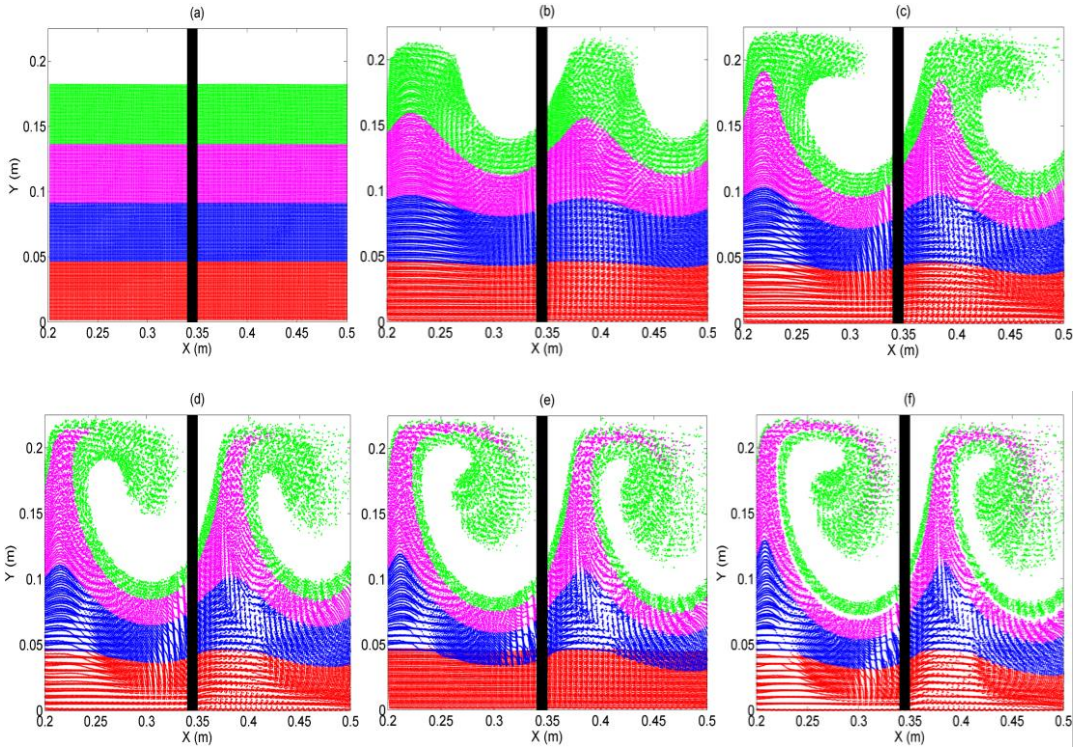


Figure 5.19 Particle advection in a pair of adjacent side-wall cavities for U_{i2} = 475 mm/s: (a) $t = 0.0s$, (b) $t = 5s$, (c) $t = 10s$, (d) $t = 15s$, (e) $t = 20s$, and (f) $t = 25s$.

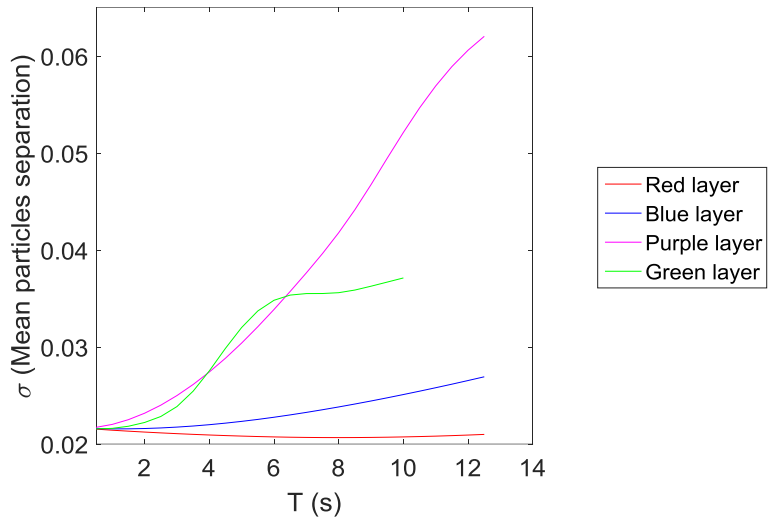


Figure 5.20 Mean particles separation for particle advection in a pair of adjacent side-wall cavities for U_{i2} = 475 mm/s.

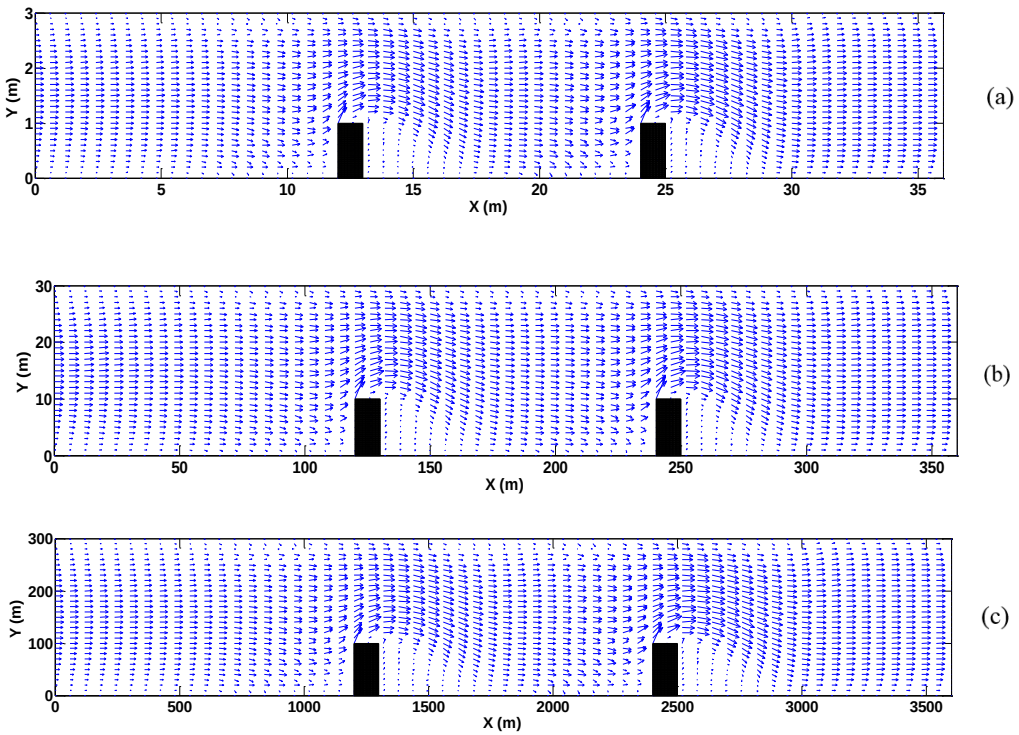


Figure 5.21Velocity vectors for open channel flow past two groynes: (a) Case *R*1, *cf* = 50; (b) Case *R*2, *cf* = 5; and (c) Case *R*3, *cf* = 0.5.

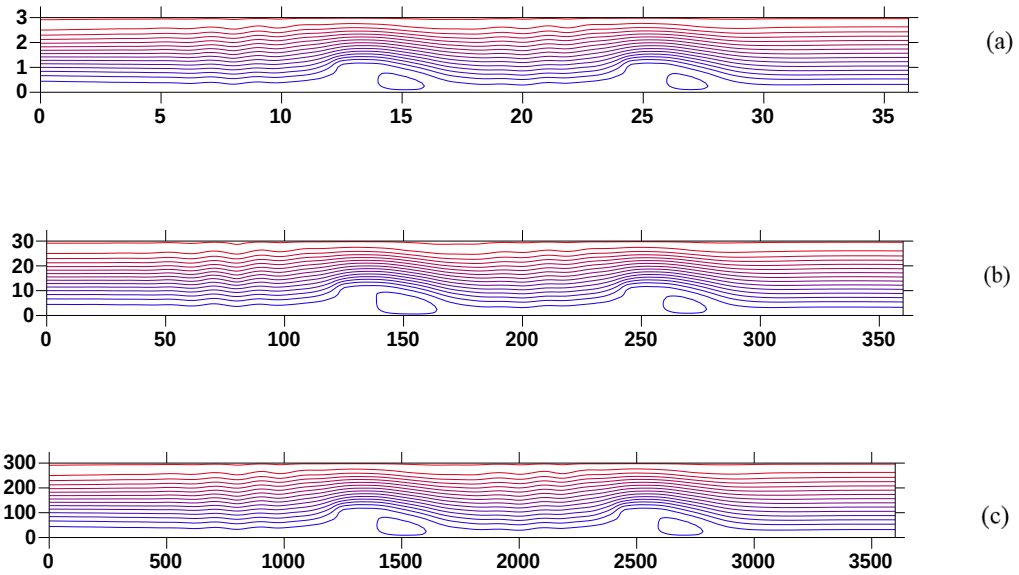


Figure 5.22 Stream function contours for open channel flow past two groynes: (a) Case *R*1, *cf* = 50; (b) Case *R*2, *cf* = 5; and (c) Case *R*3, *cf* = 0.5.

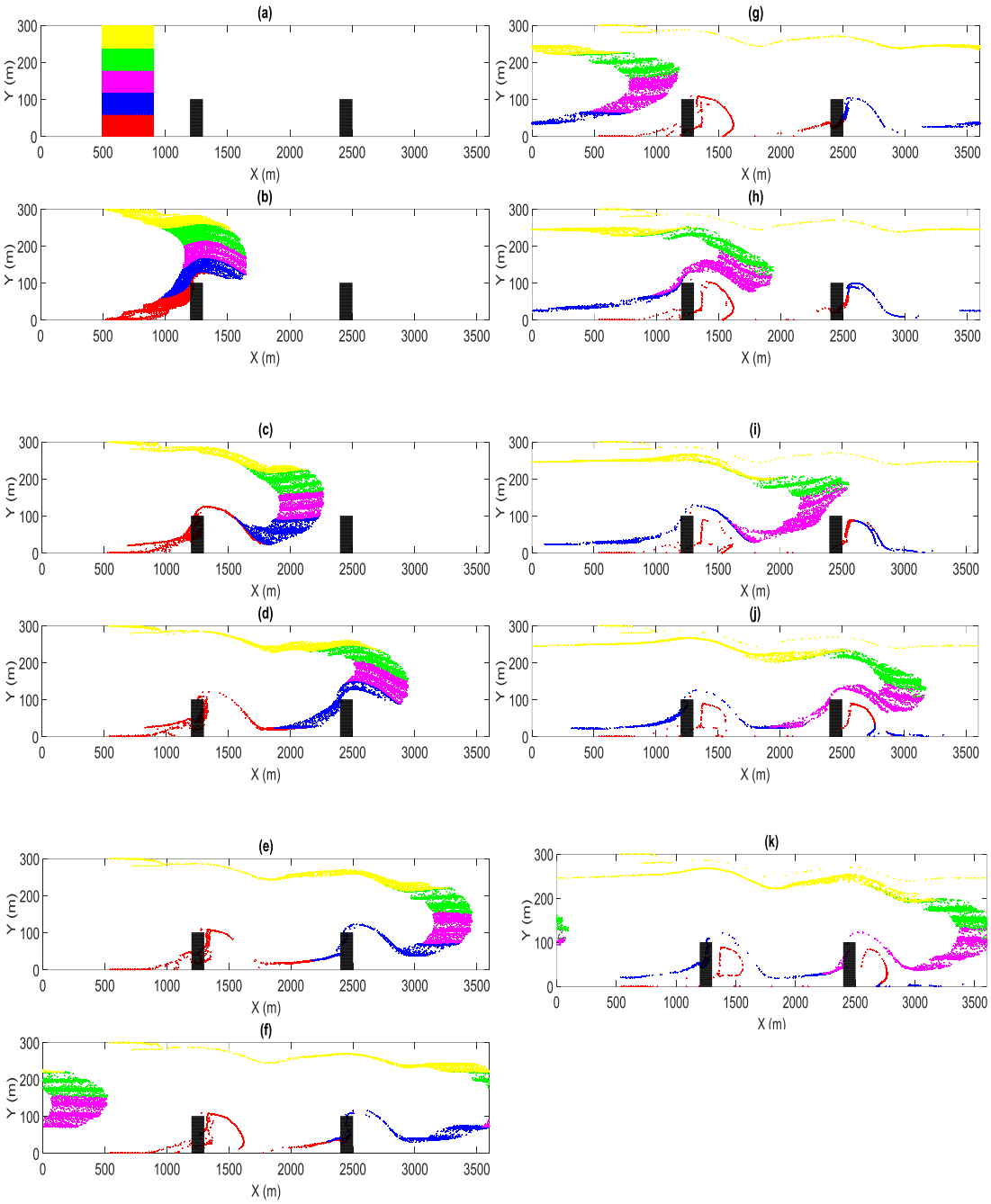


Figure 5.23 Mixing of particles in a rectangular channel containing two groynes, Case *R*3; (a) $t = 0\ s$, (b) $t = 100\ s$, (c) $t = 200\ s$, (d) $t = 300\ s$, (e) $t = 400\ s$, (f) $t = 500\ s$, (g) $t = 600\ s$, (h) $t = 700\ s$, (i) $t = 800\ s$, (j) $t = 900\ s$, and (k) $t = 1000\ s$.

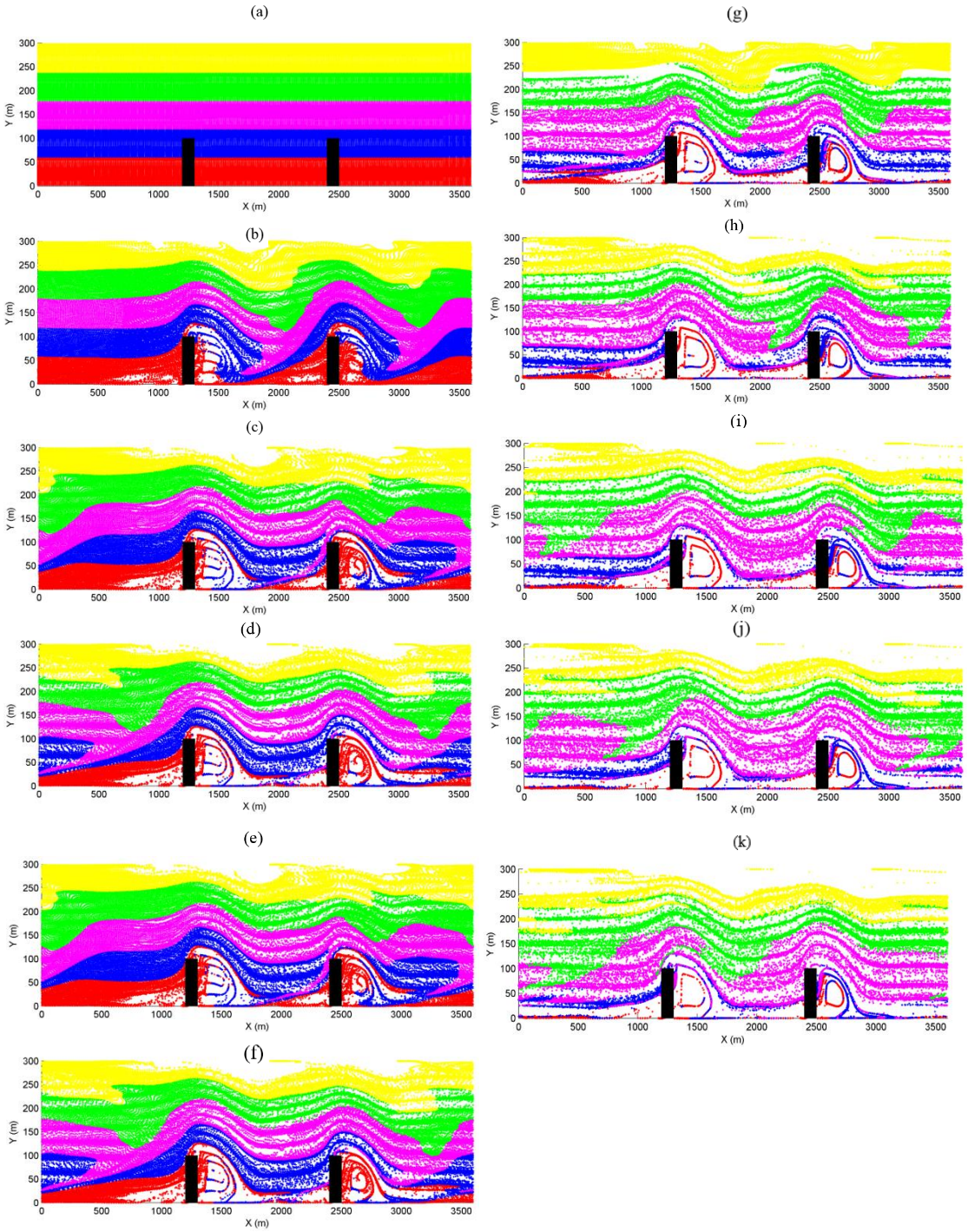


Figure 5.24 Mixing of particles in a rectangular channel containing two groynes, with particles initially filling the domain, Case *R*3; (a) $t = 0\ s$, (b) $t = 100\ s$, (c) $t = 200\ s$, (d) $t = 300\ s$, (e) $t = 400\ s$, (f) $t = 500\ s$, (g) $t = 600\ s$, (h) $t = 700\ s$, (i) $t = 800$, (j) $t = 900\ s$, and (k) $t = 1000\ s$.

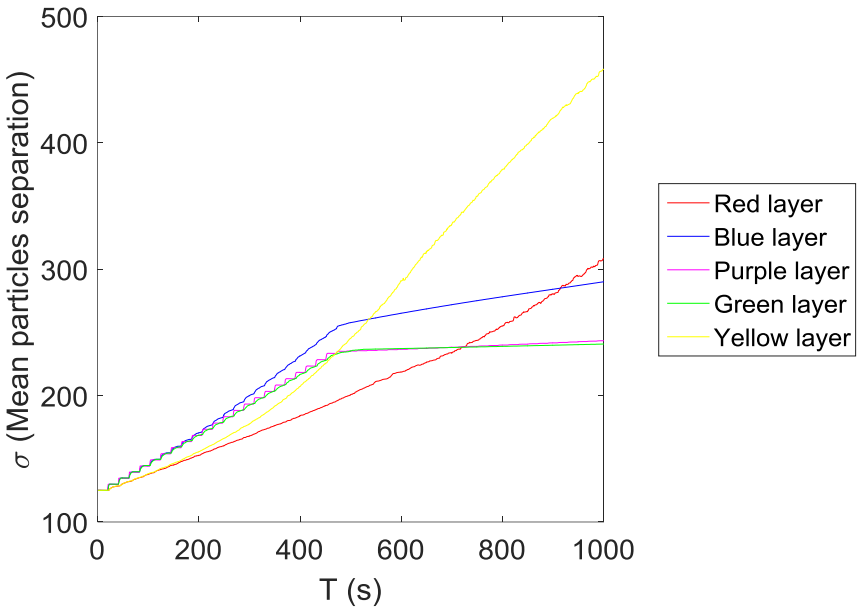


Figure 5.25 Mean particles separation as a function of time for a rectangular channel containing two groynes and initially filled with particles, Case *R*3.

Chapter 6

Case Study: Particle Advection in Idealized Danube River Bend

This chapter applies the coupled shallow water and Lagrangian particle tracking models to cases of particle advection along a shallow river reach that is an idealization of a typical bend in the Danube River, Hungary. First, the depth-averaged velocity field is determined using a curvilinear shallow water equation solver. Next, a Lagrangian particle-tracking scheme computes individual particle trajectories, with linear interpolation used to provide an estimate of the continuous velocity field from the discrete values provided by the curvilinear shallow flow solver. Mixing processes are studied through interpreting Lagrangian particle motions in a curved river bend containing groynes, whose dimensions are chosen to mimic an Idealized reach of the River Danube.

6.1 Hydrological characteristics of the Danube Basin

The River Danube is one of the largest rivers in Europe, and flows through Germany, Austria, Slovakia, Hungary, Croatia, Serbia, Romania, Bulgaria, Moldova, and Ukraine (see e.g. Pinka, 2015). The river has a total length of 2850 km and carries an annual mean discharge of 6,855 $m^3 s^{-1}$. The catchment area is 800,000 km^2 and the Danube water resource is shared by more than eighty million people. This river has its source in the Black Forest Mountains and drains into the Black sea (see e.g. Brilly *et al*. 2010). Figure 6.1 shows a satellite view of the River Danube and its catchment. According to Brilly *et al*. (2010), the Danube catchment is used by 18 countries and 10.55% of the population of Europe lives close to this catchment. Romania and Hungary have the greatest share of the Danube with respectively (28.43%) and (11.39%) of catchment.

The alpine part of the Danube experiences the largest precipitation at 3200 mm per annum rainfall whereas the lowland at the Black Sea has the minimum precipitation at 350 mm per annum. The alpine regions contain 29 lakes. However, the largest lake in the Danube catchment is Lake Balaton which has a surface area of 589 km^2 and is located in the middle Danube region (Brilly *et al.*, 2010). Maximum duration of snow cover (100 days) is located in the mountainous alpine regions. However, the Black Sea region of the Danube catchment has a minimum duration of snow cover of 9–12 days per annum. The temperature varies considerably along the Danube. The middle and lower Danube lowlands are warmest, with annual average temperature of about +12°C. The Austrian Alps are the coldest region of the catchment with annual average temperature of about -6.2°C. Consequently, the lowest rate of evaporation occurs in the alpine region (100 mm per annum), whereas the highest rate occurs at the Danube delta (400 mm per annum) (Brilly *et al*. 2010). Despite the undoubted benefits of the Danube as a source of water and hydropower, the river is prone to flooding as it passes through Central Europe. The Danube flood season occurs in spring, between March and April, when rainfall and snowmelt reach their maximum levels. In 2002, the Danube experienced its worst flood in 50 years, causing more than 100 fatalities and more than £14 billion in damage. Figure 6.2 contains satellite images acquired by NASA's Terra spacecraft before and after the Danube flooded parts of Hungary. The morphology of the Danube is an important factor in its flow characteristics, sediment carrying capacity, and vulnerability to flooding. NASA's Earth Observatory data show that the most curved boundaries of the river are located in the Hungarian part of Danube. The image in Figure 6.3 was acquired by NASA's Terra satellite on 8th April, 2006 (see NASA earth observatory website, 2015)[1] . The curved boundaries and morphological features of the Danube have an important influence on the river flow, which must be considered by the engineers concerned with Danube river management in Hungary. Figure 6.4 shows the Danube as it flowed north of Budapest near the city of Vác, Hungary before and after the 2002 flood. Severe inundation is evident, with high levels of sediment loading after the flood occurred.

[1] http://www.earthobservatory.nasa.gov

Figure 6.5a shows the flood risk map for the Danube River catchment, obtained from the International Commission for Protection of Danube River website (2015)[1] . The corresponding map of flood plains is shown in Figure 6.5b. The flood plans has key roles for agricultural lands and Ecology in Danube basin. The flood risk management maps in 2010 had anticipated Hungry and Croatia new floods which occurred in June 2013 and May 2014 causing £5 billion damage. The cost of flood inundation as the Danube curves is of great importance to stakeholder countries, such as Hungary. Flood protection systems comprising hydraulic structures is an option used in Hungary to reduce flood damage in the Danube basin. These systems are predominantly based on dykes, groynes, reservoirs and dams. Along the Danube there are 269 reservoirs with a total volume of 17,769 million m^3, and many channels, dams, diversions, intakes and flow-guiding structures. The first flood defense system was built in the 19th C to protect rural and agricultural land and comprised drainage systems and intakes aimed at collecting excess rainfall and releasing it slowly to the river, thus diminishing the amplitude of any flood wave (Brilly *et al*. 2010). These hydraulic structures also acted to improve navigation in the Danube. In the 20th C, the main channel was improved by adding sluice gates, raising levees, and providing cut off systems for tributaries. For river management purposes, the Danube has been divided into three important parts: the Upper Danube (13% of the Danube), the Middle Danube (58% of the Danube), and the Lower Danube (29% of the Danube). With regard to the flow regime, the Upper and Middle Danube can be divided to three sub-categories: high water, mean water and low water (Brilly *et al*. 2010). Most flood protection schemes were developed to mitigate against high water events in the Danube. Structures were also been constructed to improve navigation during the steady-state medium and low flow regimes. The existing primary flood protection systems have gradually became problematic because of their effect on navigation and their high cost of maintenance, and so a new generation of flood protection systems was built at the beginning of the 21st C mainly comprising riverbank reinforcement, flow-guidance structures (such as dykes, groynes, drops and chutes), and flood levees (Brilly *et al*. 2010). Figure 6.6 indicates the locations of new flood protection systems in 2015 along the entire length of the Danube (International Commission for Protection of Danube River website,

[1] http://www.icpdr.org

2015). Of the various hydraulic structures that have been added to the Danube, groynes and dykes play the dominant part in flood protection, in particular in the Middle and Lower Danube regions. These structures control the flow and stabilize the river banks so that the energy of a flood wave is reduced in the channel (Brilly *et al*. 2010).

6.2 Groyne and dyke designs

Groynes and dykes are important flood protection works along the curved boundaries of Danube. Groynes dissipate energy and dykes act as high embankments holding the flood flow within the floodplain. Novak *et al*. (2007) provide a detailed description of such hydraulic structures. Figure 6.7 shows an idealized view of the primary flow features at a river bend with groynes present. The flow separates at the ends of the groynes, and recirculation zones containing relatively slow moving water form between and behind groynes. Sediment particles tend to accumulate in these recirculation zones, and it is common practice that the groyne length is selected on the basis of its impact on sediment transport and hence river erosion. The groyne length also depends on soil texture and river navigation constraints. Along the Danube bends, the spacing between groynes is typically twice their individual length. Groynes may be arranged in straight or zigzag configurations, according to river flow control requirements. Dykes are installed along the banks of the river to prevent flood overtopping and control the flow to reduce erosion, Novak *et al*. (2007). This chapter next considers the simulation of shallow flow around a circular river bend that is an idealized version of a Danube bend. The flow patterns are simulated by solving the non-orthogonal curvilinear shallow water equations (SWEs) on a grid created using the multi-grid solver. The effect of the presence of the groynes is modelled by representing them as locally enhanced bed roughness. Lagrangian particle tracking is used to assess the advection and mixing of particles in the vicinity of the groynes.

6.3 Simulation of Idealized Danube River Bend

The numerical model is now extended to consider flow in a circular bend representing an idealization of a typical Danube River bend. Here, the river bend

geometry is mapped onto a curvilinear (polar) system, and the transformed shallow water equations solved.

6.3.1 Case 1, 90° Danube River bend

The first physical domain to be considered is a 90° circular bend, where the width of the river is b = 500 m. The bend has inner radius r_0 = 1000 m and outer radius r_1 = 1500 m. To implement the curvilinear method, the physical Cartesian (x, y) domain is mapped onto a rectangle in the transformed (polar) (r, θ) domain, as indicated in Figure 6.8a,b. Borders S_1 , S_2 , S_3 and S_4 are mapped onto T_1, T_2 , T_3 and T_4 , respectively, and the interior grid generated using the multi-grid method (see Section 2.3). Figures 6.8c,d show the resulting 120 x 120 grid in the physical and transformed systems.

By definition,

$$r = \sqrt{x^2 + y^2} \qquad (6.1)$$

and

$$\theta = \tan^{-1}(\frac{y}{x}),\ \cos\theta = \frac{x}{r},\ \sin\theta = \frac{y}{r}\ . \qquad (6.2)$$

The velocity vector components in the Cartesian and polar systems are related by

$$u = u_r \cos\theta - u_\theta \sin\theta \qquad (6.3a)$$

$$v = u_r \sin\theta + u_\theta \cos\theta \qquad (6.3b)$$

where u and v are the Cartesian velocity components, and u_r and u_θ are the corresponding radial and tangential velocity components (see Figure 6.9). Rearranging equations (6.3), we obtain

$$u_r = u\cos\theta + v\sin\theta \qquad (6.4a)$$

$$u_\theta = v\cos\theta - u\sin\theta \qquad (6.4b)$$

For streamlined flow around a circular bend, u_r is zero, and equations (6.4) reduce to

$$u = -u_\theta \sin\theta\ ,\quad v = u_\theta \cos\theta \qquad (6.5)$$

The inlet velocity condition at $\theta = \frac{\pi}{2}$ (See Figure 6.8a) is:

$$u = -u_\theta\,, \quad v = 0 \tag{6.6}$$

The outlet boundary condition velocity at $\theta = 0$ is:

$$u = 0\,, \quad v = u_\theta \tag{6.7}$$

For streamlined flow around a circular bend, an analytical solution can be used to verify the shallow flow solver. Here, a parabolic velocity distribution across the channel is input, corresponding to fully developed flow in a channel (Schlichting, 1968, p177) such that:

$$u_\theta = \frac{3}{2}u_I\left(1 - \left(\frac{2r'}{b}\right)^2\right)\,, \qquad u_r = 0 \tag{6.8}$$

where u_I is the mean flow speed at the inlet and r' is the distance measured radially from the midpoint of the channel. Here, the eddy viscosity in the river bends is determined by using the depth-averaged eddy viscosity formula.

$$v = 5.9h\frac{\sqrt{g\left(U_\theta^2 + U_r^2\right)}}{C} \tag{6.9}$$

Where, $C = \frac{R^{\frac{1}{6}}}{n} = 40$. Applying equation (6.9), the eddy viscosity is 9.3 m^2/s. Table 6.1 lists the other physical parameters governing the system that are input to the numerical model for shallow flow around the bend with a set bed roughness, and no wind stress present. Figure 6.10 depicts the bed elevation contours obtained from;

$$z_b = 2 - \frac{s_n}{s_{max}}\,, \tag{6.10}$$

in which $s_n = b\theta$, where, $\Delta\theta = \frac{\pi}{2\,imax}(i), i$ =0,1,2,..., $imax$ =120, $\theta = \frac{\pi}{2}\Delta\theta$, and $s_{max} = \frac{\pi}{2}\,b$ where b = 500 m.

Figure 6.11 presents the inlet and outlet velocity time series showing that steady-state is achieved by t ~ 1500 s. Figure 6.12 presents predicted and analytical solutions for the steady-state velocity vectors and contours for fully developed shallow flow around

a 90^o bend of mean inlet velocity u_I = 0.889 m/s. Here, a 120 x 120 mesh has been used, along with a time step of 0.1 s. The results show selected vectors at a time, t = 5000 s. There is excellent agreement between the two solutions. Using Simpson's rule to integrate the velocity over the flow cross-section, the numerically predicted discharge at steady-state is 1433.9 m^3/s at the inlet and 1431.8 m^3/s at the outlet. The analytical value of discharge is 1333.5 m^3/s. The difference between analytical and numerical discharges indicates that the error might have happened for the numerical discharge, that is, the central difference discretization has small error in SWEs solver. Here, the error is 7%.

6.3.2 Case 2, 180^o Danube River bend

Next, consider shallow flow around a 180^o circular bend of inner radius r_0 = 1000 m and outer radius r_1 = 1500 m. Figures 6.13a,b show the physical domain mapped onto the transformed domain, and Figures 6.13c,d show the 120 x 120 grid in the physical and transformed systems. The bed elevation contours (Figure 6.14) are determined from Equation (6.10) in which $s_n = b\theta$, where $\theta = \pi\Delta\theta$, and $s_{max} = \pi\, b$ where b = 500 m. Table 6.1 lists the physical parameters. Again the time step is 0.1 s. The inlet and outlet velocity plots with time in Figure 6.15 confirm that steady state is again reached by t ~ 1500 s. Figure 6.16 shows the excellent agreement between the numerical and analytical predictions of the (selected) velocity vectors and magnitude contours at t = 5000 s. The numerical estimates of the inlet and outlet discharge are 1436.9 m^3/s and 1430.0 m^3/s in close agreement with the analytical value of discharge is 1333.5 m^3/s. Also, the discharge error is 7%.

6.3.3 Case 3, 90^o bend with inlet and outlet stems

For the case of shallow flow around a 90^o bend with inlet and outlet stems, both stems are 2000 m long and 500 m wide. The 90^o bend has inner radius r_0 = 1000 m and outer radius r_1 = 1500 m. Figure 6.17 shows the physical domain mapped to the transformed domain and the corresponding 120 x 120 grid. Table 6.1 lists the physical parameters. There is no wind stress present. Figure 6.18 displays the bed elevation contours. Figure 6.19 presents the inlet and outlet velocity time series showing that steady-state is achieved by t ~ 2500 s. The steady-state velocity vectors and velocity

magnitude contours for shallow flow around a 90^o bend with inlet and outlet stems indicate excellent agreement between the numerical and analytical predictions at t = 5000 s (see figure 6.20). The numerically predicted discharge at steady-state is 1414.4 m^3/s at the inlet and 1415.1 m^3/s at the outlet. The analytical value of discharge is 1333.5 m^3/s. the discharge error is 6.1%.

6.3.4 Case 4, 180^o bend with inlet and outlet stems

Shallow flow around a 180^o bend with inlet and outlet stems is now considered. Again, both stems are 2000 m long and 500 m wide. Figure 6.21 shows the physical and transformed domains and corresponding 120 x 120 grids. Figure 6.22 presents the bed elevation contours. The physical parameters are listed in Table 6.1. Wind stress is neglected. Figure 6.23 presents the inlet and outlet velocity time series showing that steady-state is achieved by t ~ 2500 s. The numerical and analytical predictions of the velocity vectors and velocity magnitude contours at t = 5000 s are almost identical (Figure 6.24). Last but not least, discharge at steady-state is 1373.6 m^3/s at the inlet and 1373.5 m^3/s at the outlet. The analytical value of discharge is 1333.5 m^3/s. Here, the discharge error is 3%. Comparison Discharge results for case 2 with case 4 denotes the improvement in discharge.

6.4 Particle Advection at Idealized Danube River Bend with Groynes

A 90^o river bend with inlet and outlet stems is generated using a combination of Cartesian grids (in the stems) and a curvilinear grid along the circular portion of the bend. The inlet stem is 3600 m long and 300 m wide. The bend has interior radius of 3300 m and an exterior radius of 3600 m. The outlet stem is 3600 m long and 300 m wide. The combined grid consists of a rectangular portion of 360 x 30 grid points covering the inlet stem, a 472 x 30 curvilinear grid for the circular bend, and a further 360 x 30 Cartesian grid for the outlet stem. To provide identical analysis of the inlet stem, the flow parameters are set to same parameters in the previous study of rectangular open channel containing a pair of groynes (case *R*3) in chapter 5 since the inlet stem geometry in the current study is similar to the former rectangular channel. Keeping that in mind, the inlet flow velocity is 0.5 m/s, the downstream water depth

is 1 m, and the eddy viscosity is 0.5 m^2/s. In the numerical model, the time step is 5 s, and the total simulation lasts 5000 s. The permeable groynes are inserted in the river bend as patches of locally increased bed roughness, such that cf_1 = 0.5. The bed roughness elsewhere is zero. Figures 6.25a and 6.25b show the steady state velocity vectors and streamlines in a portion of the river bend containing two parallel groynes projecting into the main channel from the interior curved channel wall. Recirculation zones occur in the lee of the groynes. To observe the effect of these recirculation zones on the trajectories taken by tracer particles, three initially square arrays of particles are released, the red array initially located immediately upstream of the first groyne, the blue array within the circulation zone downstream of the first groyne, and the green array located downstream of the second groyne (inside the recirculation zone). The time step used for tracking the particles is 5 s. Figure 6.25c shows the initial positions of the three coloured particle arrays. Figure 6.25d illustrates the distribution of the coloured particles at time t = 1500 s. It can be seen that the red particles introduced immediately upstream of the first groyne travel around the groyne and do not enter the recirculation zone behind the groyne. The red particles travel across the second groyne (owing to it being represented as a porous patch with increased bed roughness) and enter the recirculation zone in its lee. Blue and green particles are trapped by gyres downstream of the first and second groynes respectively. Further information on the particle advection and mixing processes can be gleaned from Figure 6.26 which shows the movement and dispersion of three bands of coloured particles introduced immediately upstream of the first groyne. The green particles are advected rapidly by the through-flow above the groynes, whereas the blue particles are pulled down towards the bed as the flow passes over the recirculation zone. The red particles creep across the first groyne and some are trapped within the recirculation zone. Figure 6.27 shows similar results obtained when the colour bands are introduced between the first and second parallel groynes.

Figures 6.28 to 6.30 show the corresponding results obtained when the groynes are arranged in an alternating configuration. Figure 6.28 shows the steady state velocity vectors and stream function contours, along with the initial (t = 0 s) and final (t = 650 s) positions of three arrays of red, green, and blue particles. The flow again separates at each of the groynes, leading to recirculation zones behind, which serve to

trap particles. Figures 6.29 and 6.30 show further visualizations of particle distributions for coloured arrays arranged in bands initially in front of the first groyne and between the two groynes (but along the inner bend). In this case of alternate groynes, the particles are not trapped for as long as for the parallel groynes and tend to be washed downstream (except for those closest to the side walls).

6.5. Conclusions

A primary reason for providing flood protection systems along rivers is to reduce the flow speed and dissipate energy in the main channel through flow control devices. Here, shallow flow around a circular bend with and without inlet and outlet stems has been considered, the geometry representative of a Danube River bend. Computational grids have been created using the multi-grid curvilinear grid generator described in Chapter 2, and the flow hydrodynamics computed by solving the transformed shallow flow equations. Groynes have been introduced by means of increased local bed roughness. The Lagrangian particle tracker has been used to study advection and mixing as particles are transported past groynes located in parallel and staggered configurations.

Table 6.1: Parameters for shallow flow in the river bends (dimensions based on idealised River Danube)

h_0 (m)	Chézy C ($m^{1/2}s^{-1}$)	ρ (kg/m^3)	g (m^2/s)	υ (m^2/s)
3.0	40.0	1000.0	9.81	9.3

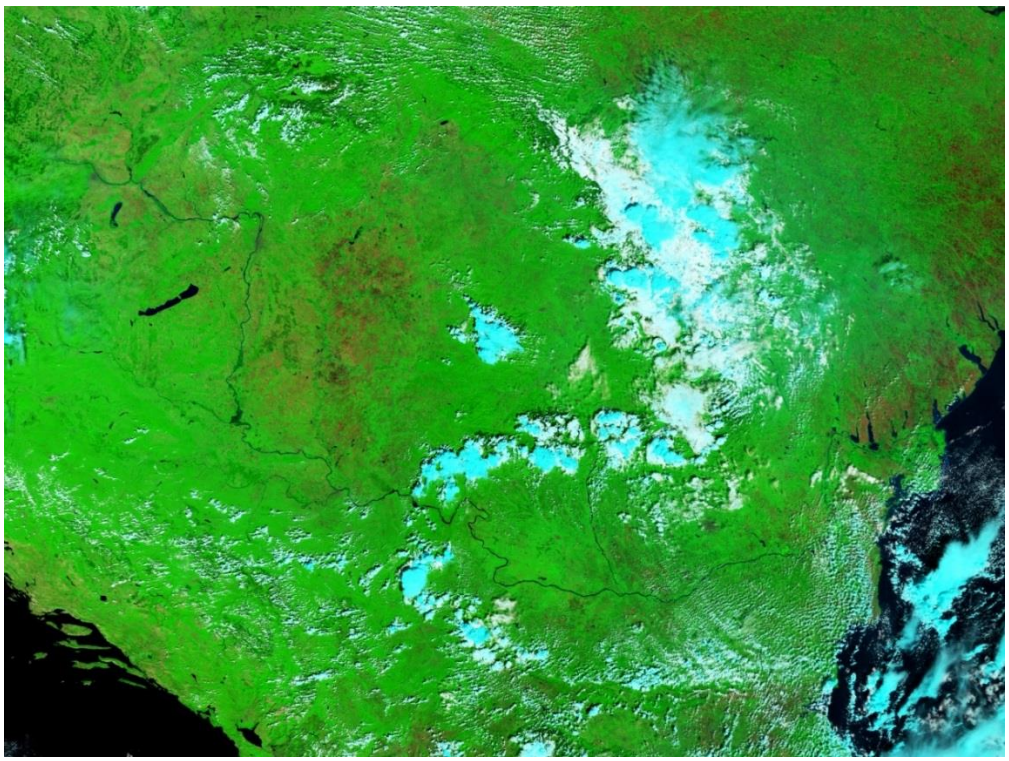


Figure 6.1 NASA's Terra satellite picture of Danube, obtained from NASA Earth observatory website (eoimages.gsfc.nasa.gov/images/imagerecords/10000/10062/danube.TMO2002231_lrg.jpg)

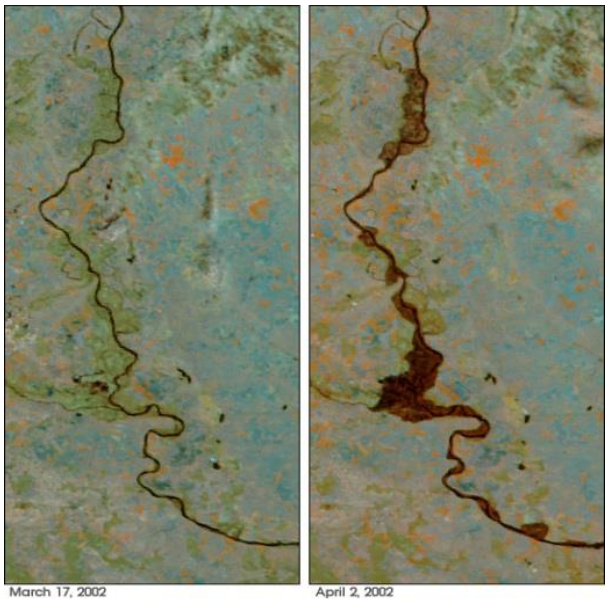


Figure 6.2 Flood in Danube in 2002 obtained from NASA Earth observatory website (http://earthobservatory.nasa.gov/NaturalHazards/view.php?id=9374)

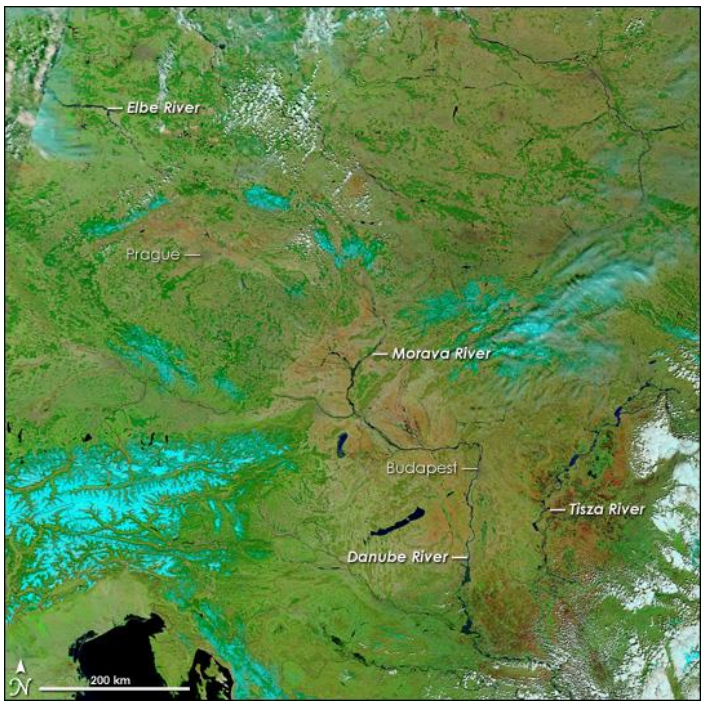


Figure 6.3 Hungarian part of Danube obtained from NASA Earth observatory website (http://earthobservatory.nasa.gov/NaturalHazards/view.php?id=16361)

(a) (b)

Figure 6.4 Danube river bend located near the city of Vác, Hungary: (a) before 2002 flood [photograph by Péter Kesselyák, obtained from panoramio.com (http://www.panoramio.com/photo/2929740)], (b) on 18th August 2002, after the flood [NASA Earth observatory website (http://earthobservatory.nasa.gov/IOTD/view.php?id=2743)]

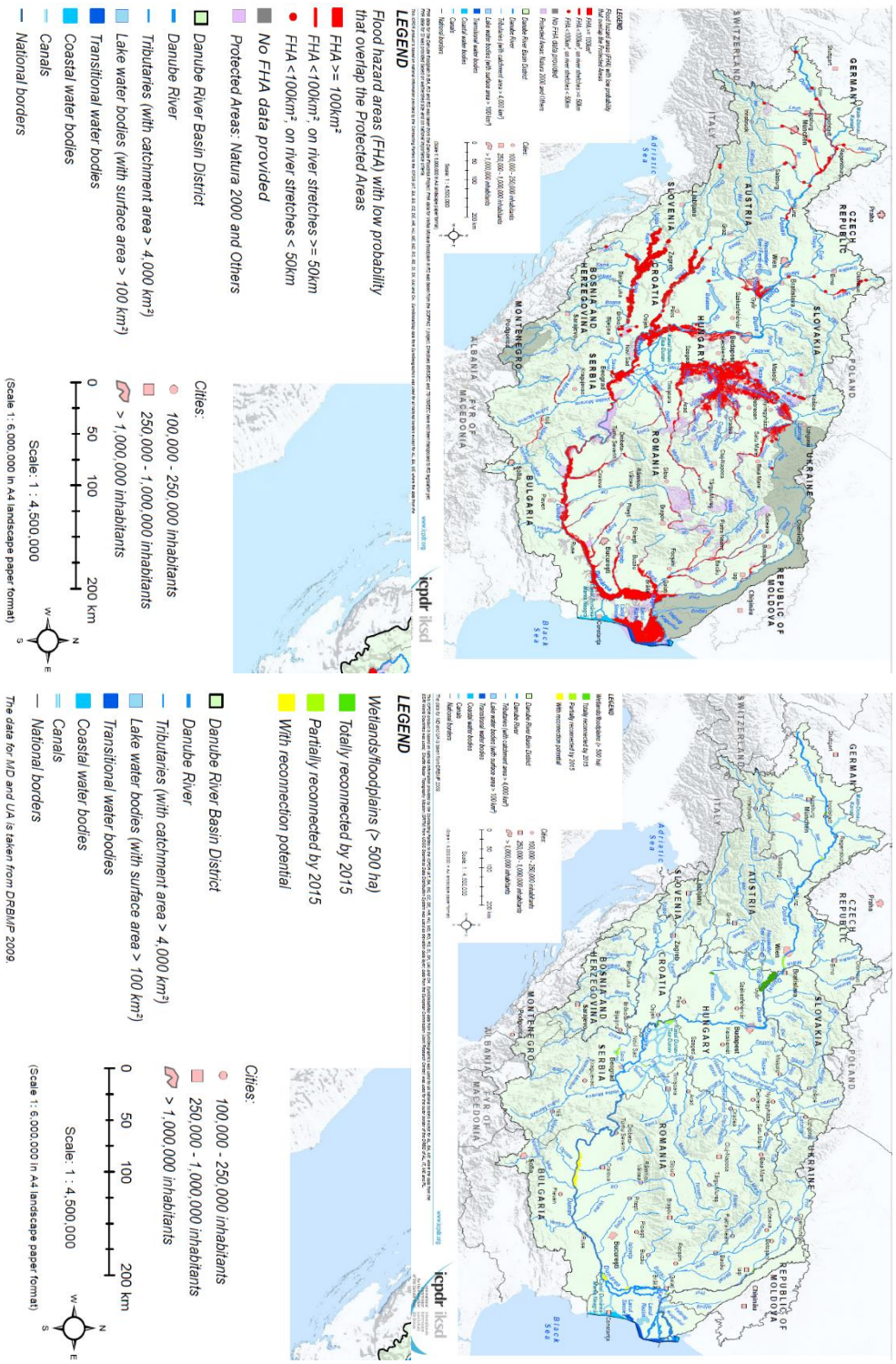


Figure 6.5 Danube river basin: (a) map of potential flood risk areas, obtained from International Commission for Protection of Danube River, ICPDR. (www.icpdr.org/main/publications/maps-danube-flood-risk-management-plan-2015), (b) map of flood plains, obtained from ICPDR (http://www.icpdr.org/main/publications/maps-danube-river-basin-district-management-plan-2009)

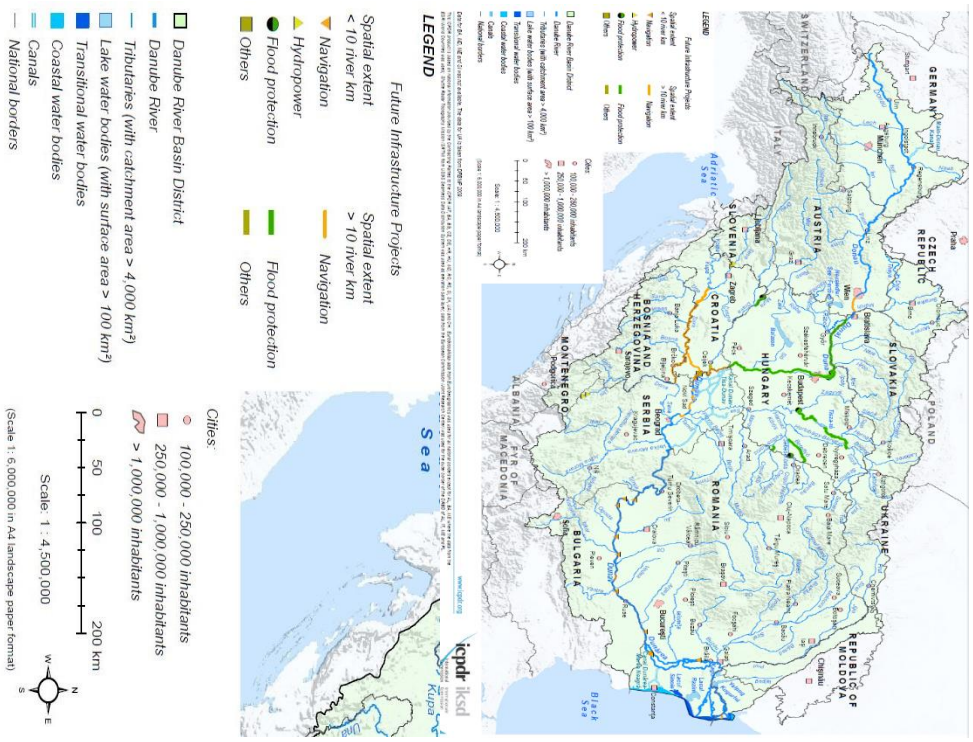


Figure 6.6 Danube river basin: new flood protection systems in 2015, obtained from ICPDR (http://www.icpdr.org/main/sites/default/files/nodes/documents/drbm_plan_-_update_2015_-_maps__dec_2014.pdf)

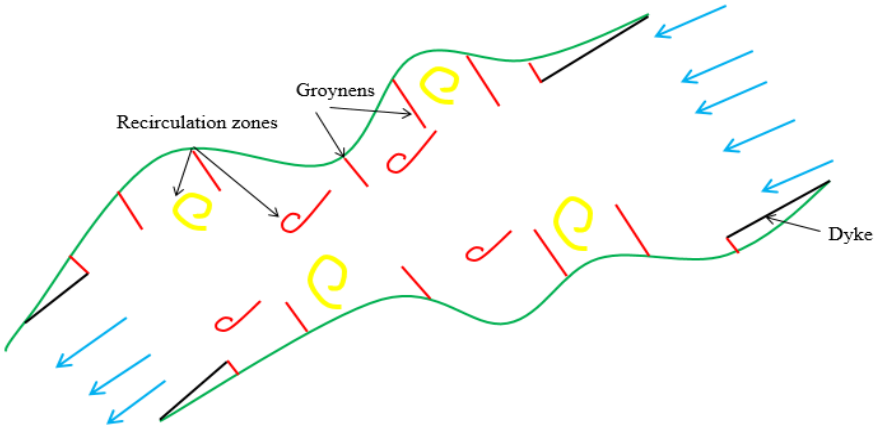


Figure 6.7 Position of groynes and dykes in a typical river

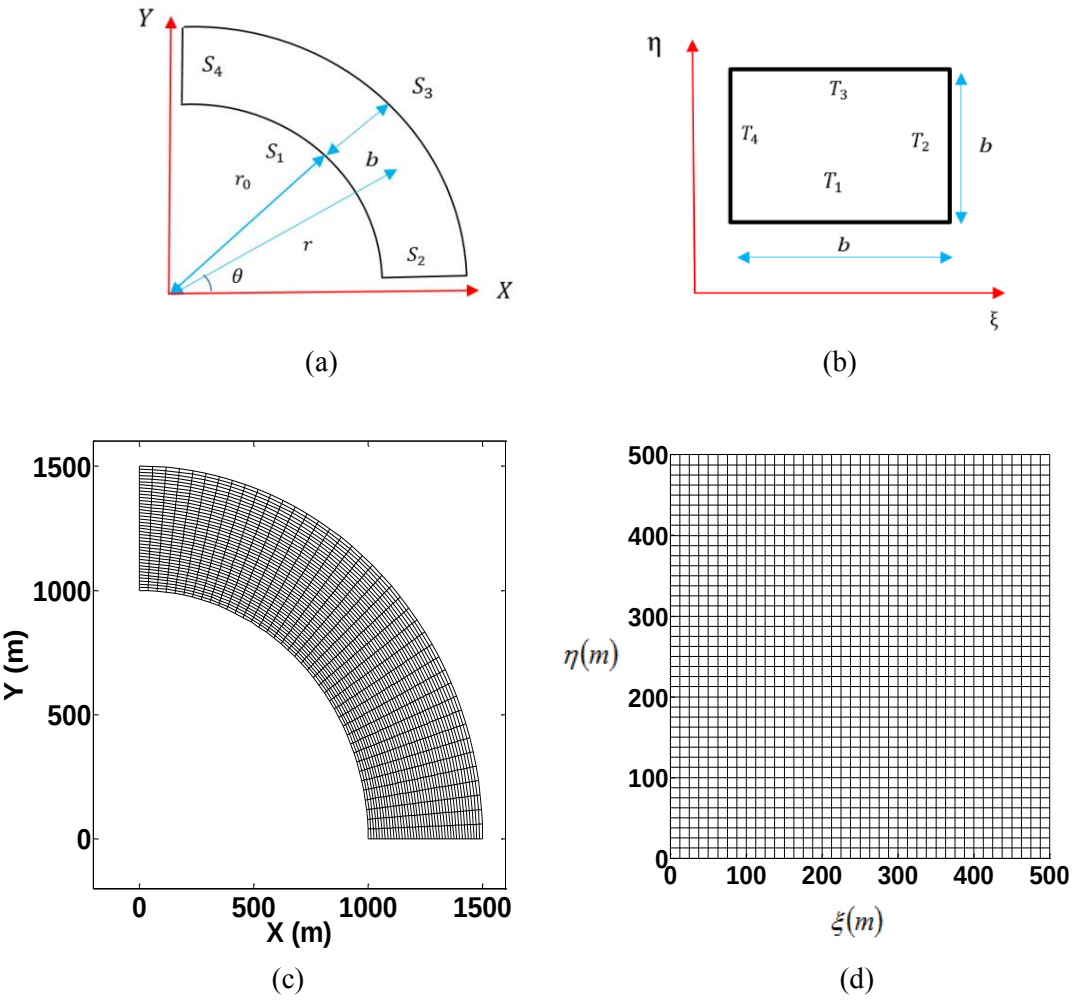


Figure 6.8 Shallow flow around a 90° bend (dimensions based on idealised River Danube): (a) physical domain; (b) transformed domain; (c) 120 x 120 physical grid; (d) 120 x 120 computational grid in transformed domain.

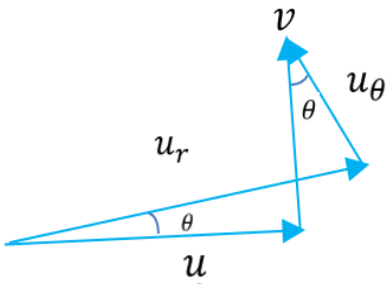


Figure 6.9 Relationship between Cartesian and polar velocity vector components

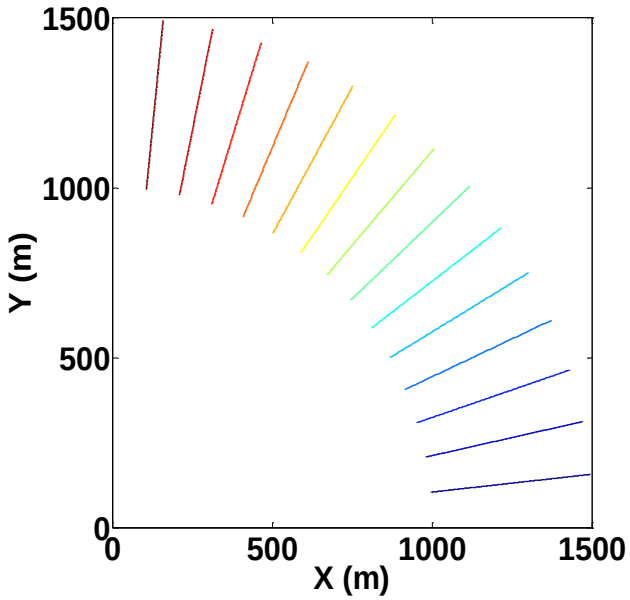
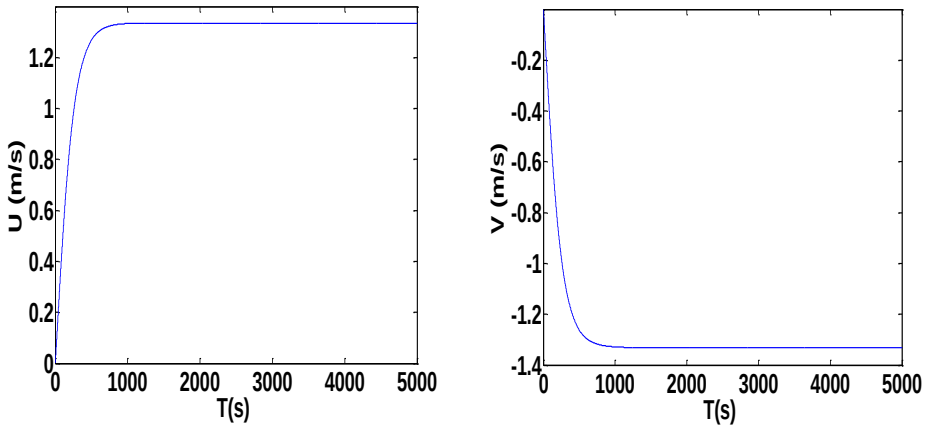
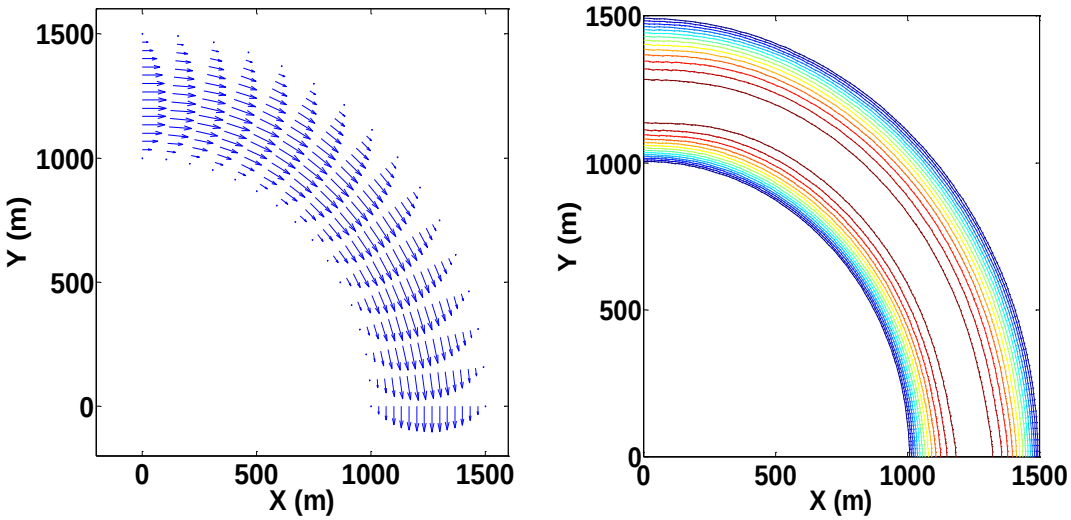


Figure 6.10 Shallow flow around a 90º bend (dimensions based on idealised River Danube): bed elevation contours.

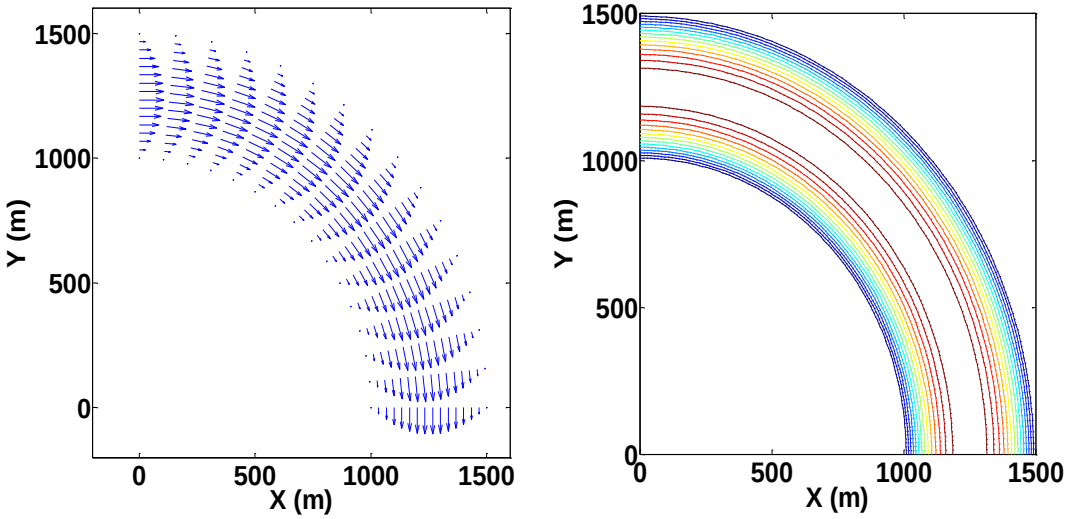


(a) (b)

Figure 6.11 Shallow flow around a 90º bend (dimensions based on idealised River Danube): (a) inlet *u*-velocity component; and (b) outlet *v*-velocity component time histories.



(a)



(b)

Figure 6.12 Steady-state velocity vectors and magnitude contours for shallow flow around a 90^{o} bend (dimensions based on idealised River Danube): (a) predicted; and (b) analytical results.

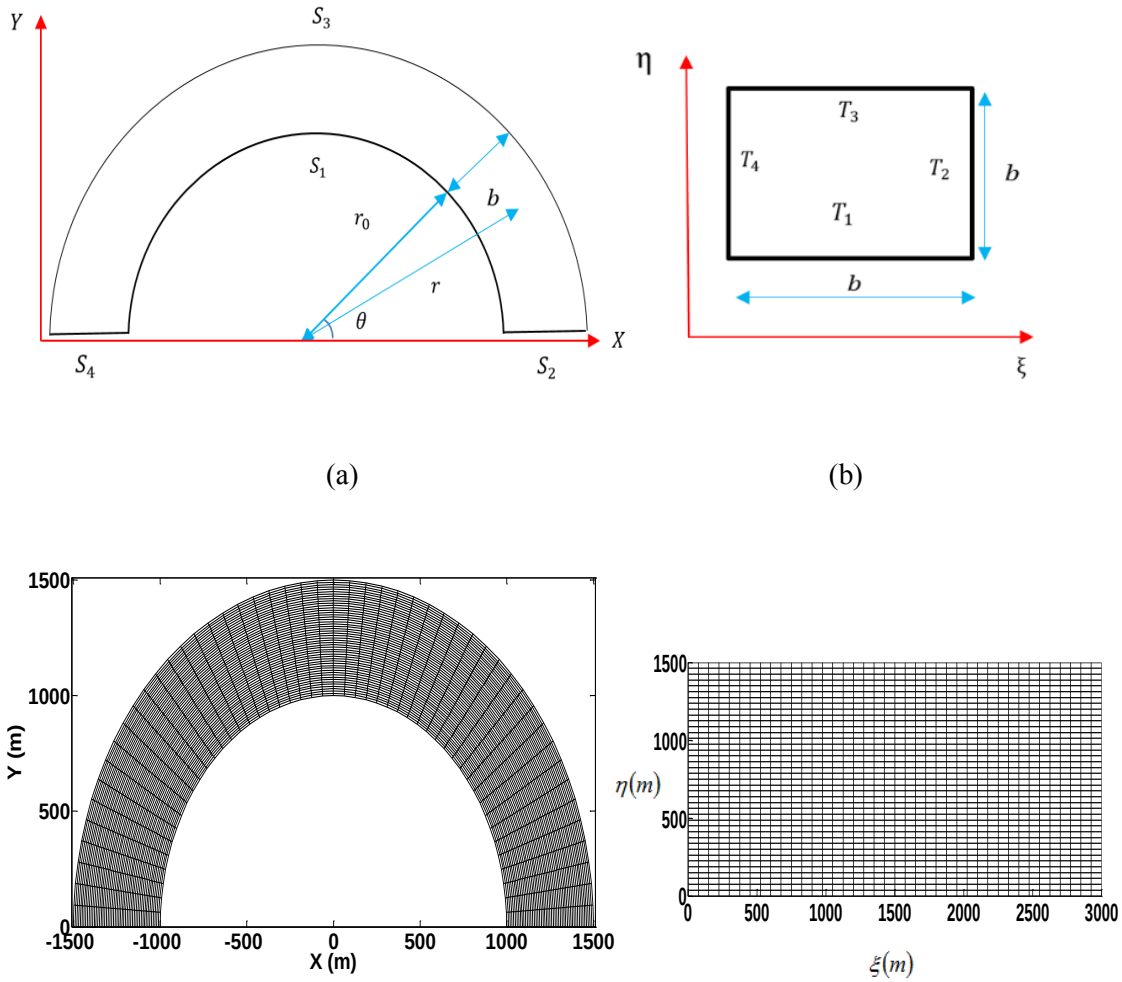


Figure 6.13 Shallow flow around a 180^{o} bend (dimensions based on idealised River Danube): (a) physical domain; (b) transformed domain; (c) 120 x 120 physical grid; (d) 120 x 120 computational grid in transformed domain.

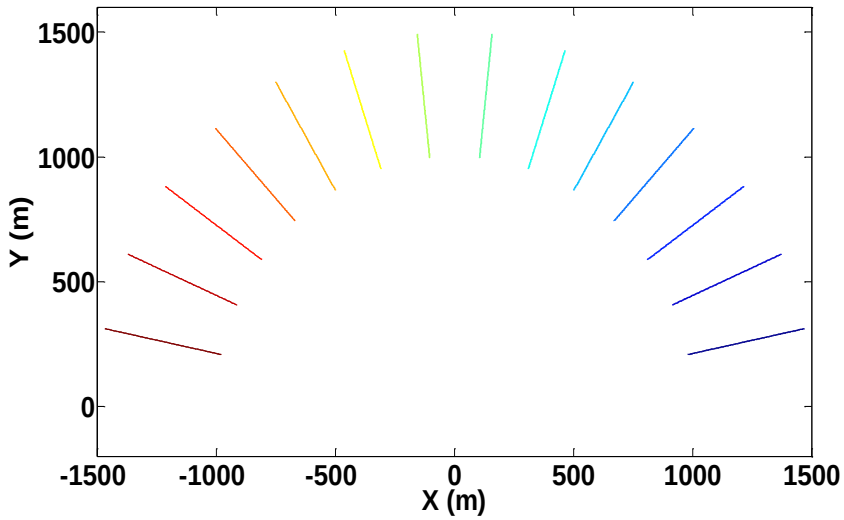
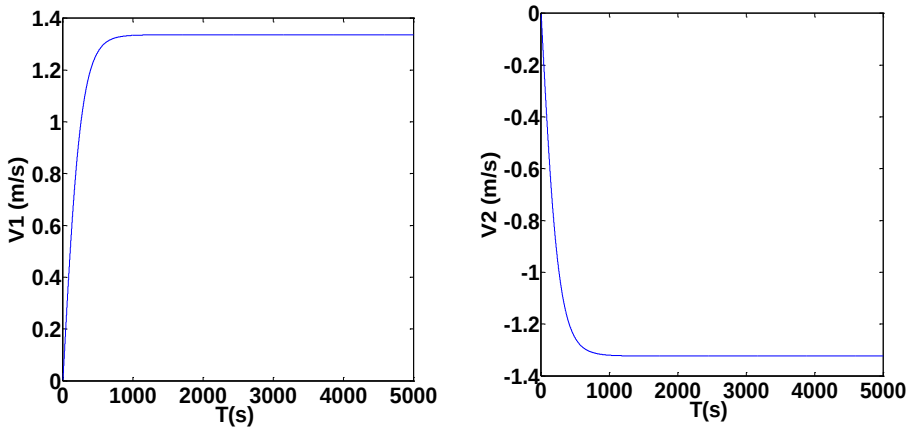
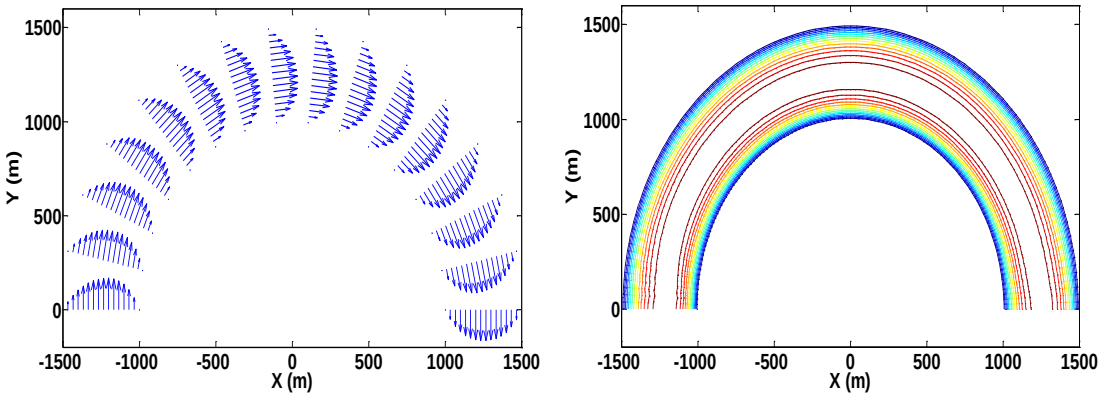


Figure 6.14 Shallow flow around a 180º bend (dimensions based on idealised River Danube): bed elevation contours.

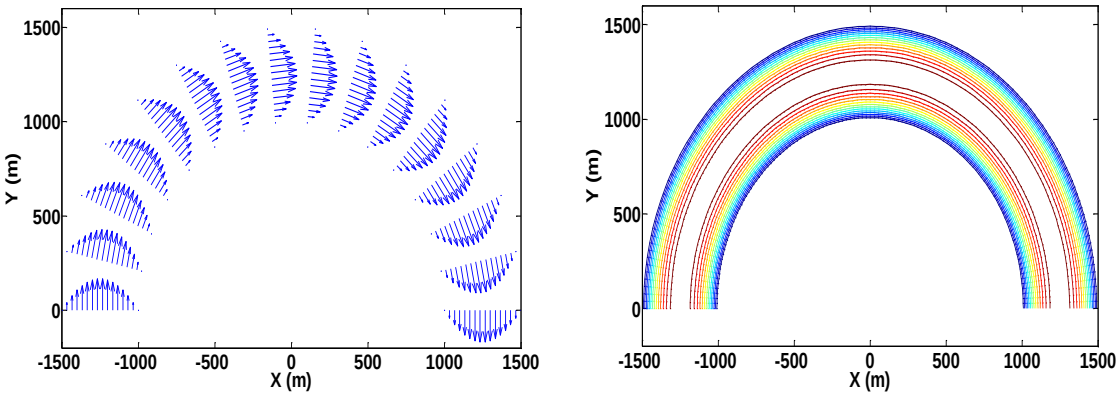


(a) (b)

Figure 6.15 Velocity time histories for shallow flow around a 180º bend (dimensions based on idealised River Danube): (a) inlet v_1-velocity component; and (b) outlet v_2-velocity component time histories.

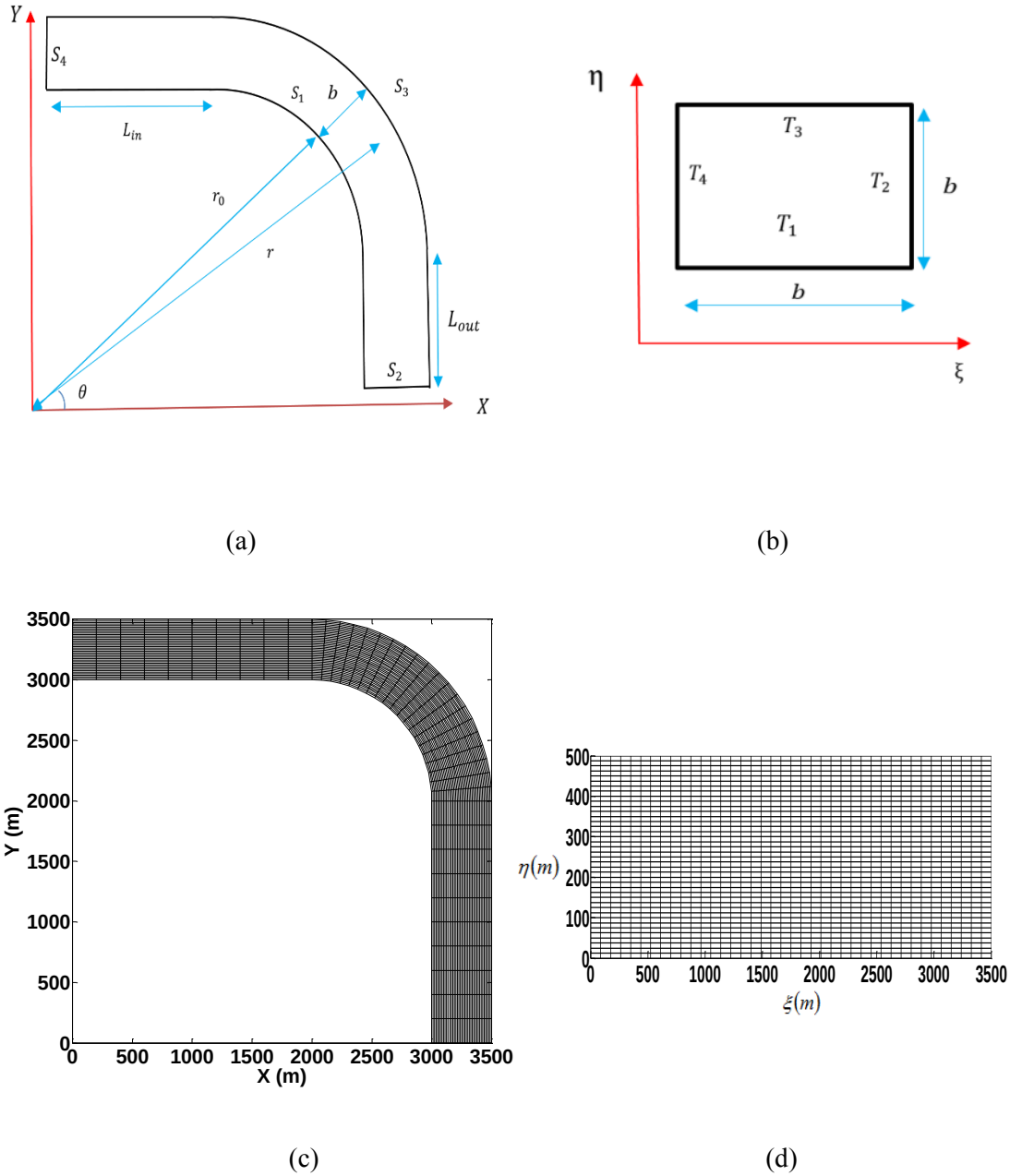


(a)



(b)

Figure 6.16 Steady-state velocity vectors and velocity magnitude contours for shallow flow around a 180^{o} bend (dimensions based on idealised River Danube): (a) predicted; and (b) analytical results.



(a) (b)

(c) (d)

Figure 6.17 Shallow flow around a 90° bend with inlet and outlet stems (dimensions based on idealised River Danube): (a) physical domain; (b) transformed domain; (c) 120 x 120 physical grid; (d) 120 x 120 computational grid in transformed domain.

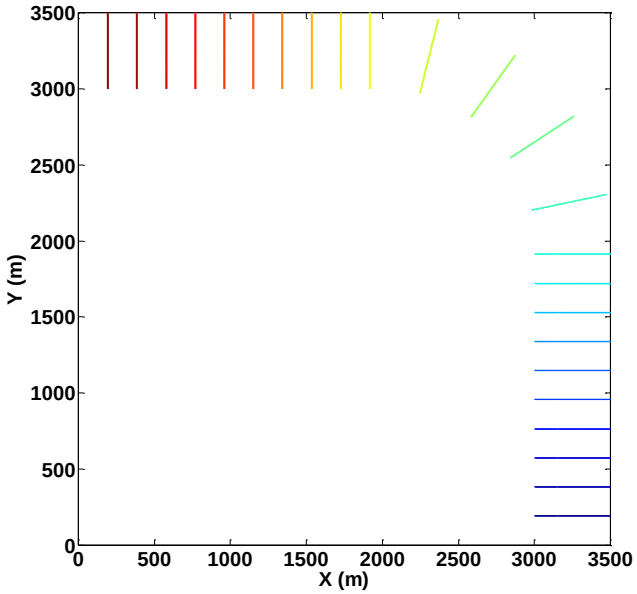
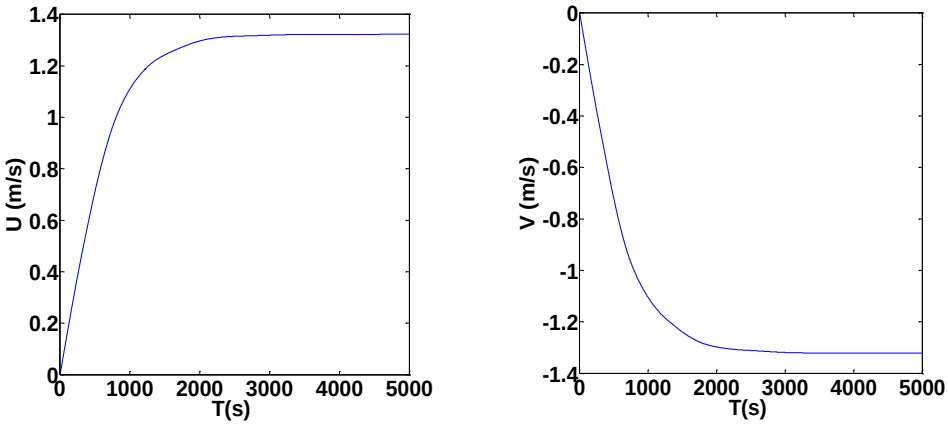
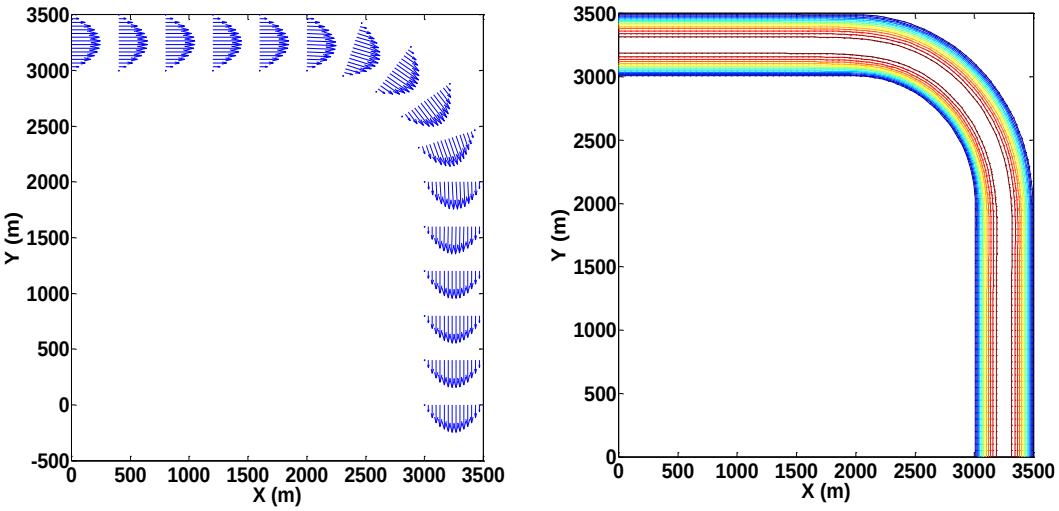


Figure 6.18 Shallow flow around a 90° bend with inlet and outlet stems (dimensions based on idealised River Danube): bed elevation contours.

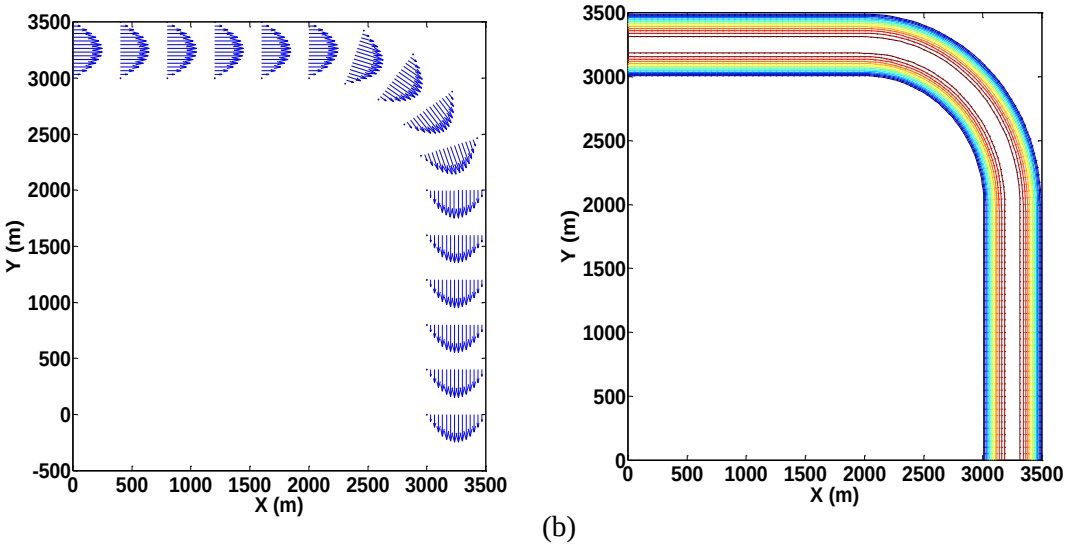


(a) (b)

Figure 6.19 Velocity time histories for shallow flow around a 90° bend with inlet and outlet stems (dimensions based on idealised River Danube): (a) inlet *u*-velocity component; and (b) outlet *v*-velocity component time histories.

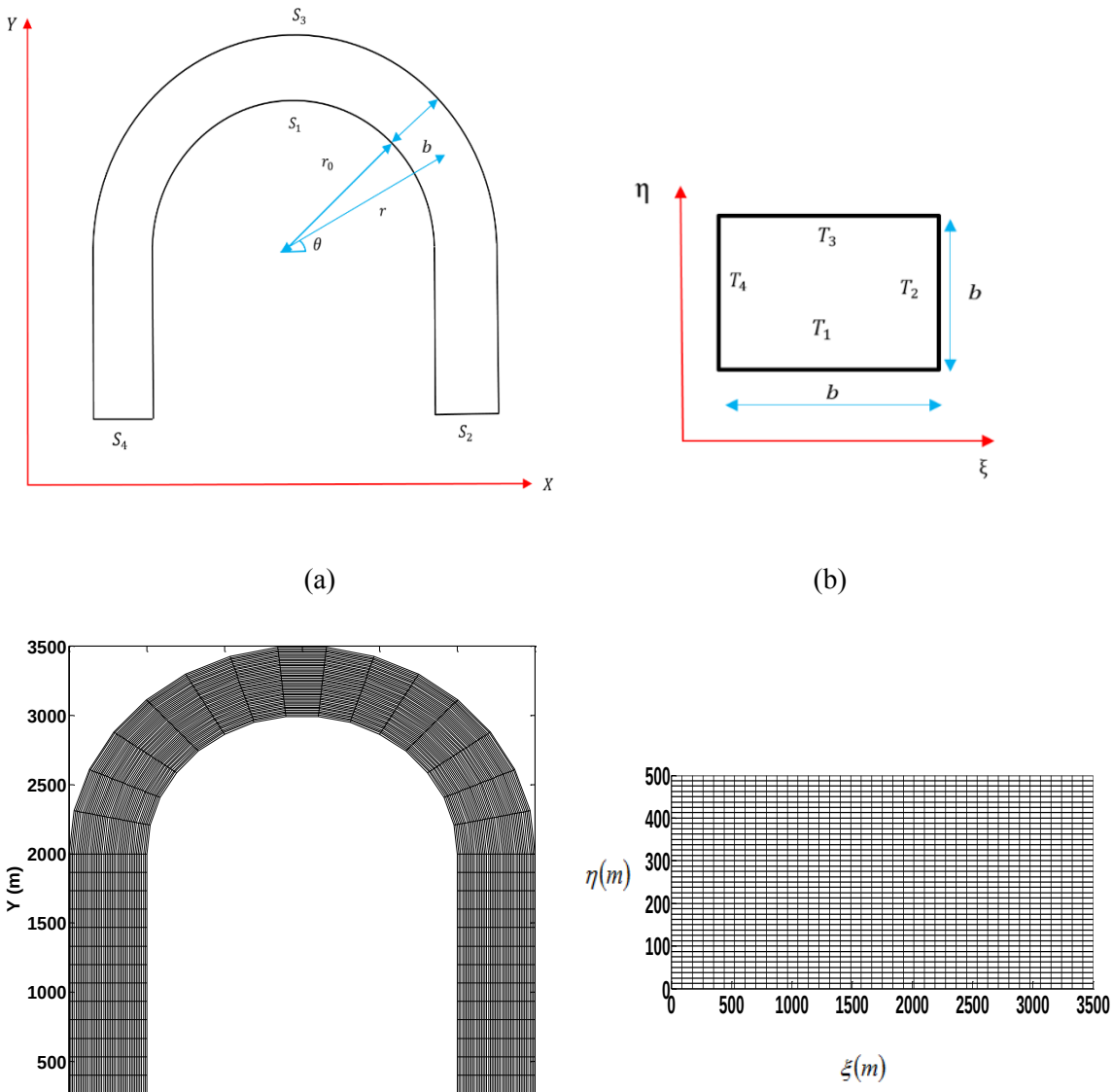


(a)



(b)

Figure 6.20 Steady-state velocity vectors and velocity magnitude contours for shallow flow around a 90º bend with inlet and outlet stems (dimensions based on idealised River Danube): (a) predicted; and (b) analytical results.



(a) (b)

(c) (d)

Figure 6.21 Shallow flow around a 180^o bend with inlet and outlet stems (dimensions based on idealised River Danube): (a) physical domain; (b) transformed domain; (c) 120 x 120 physical grid; (d) 120 x 120 computational grid in transformed domain.

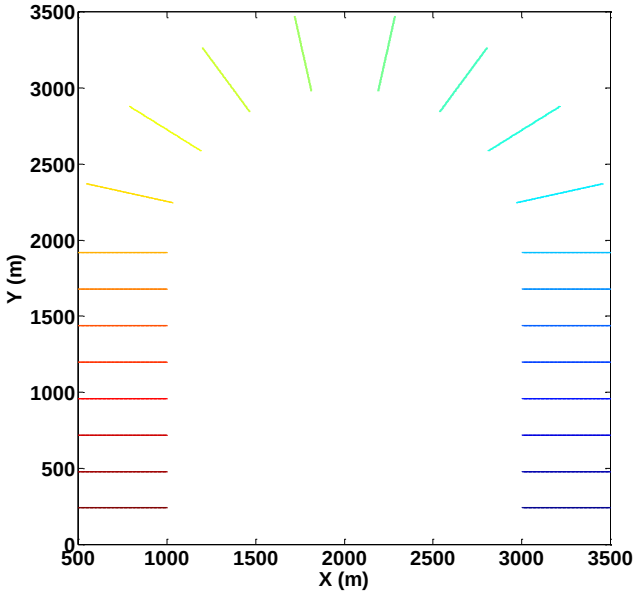
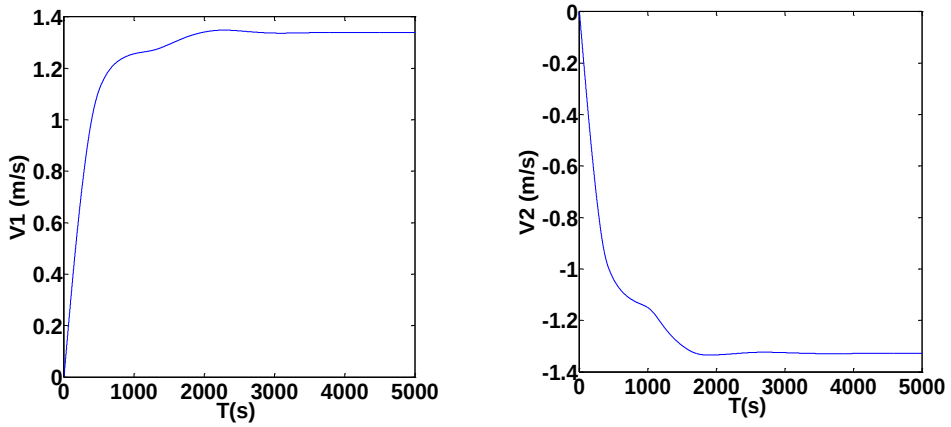
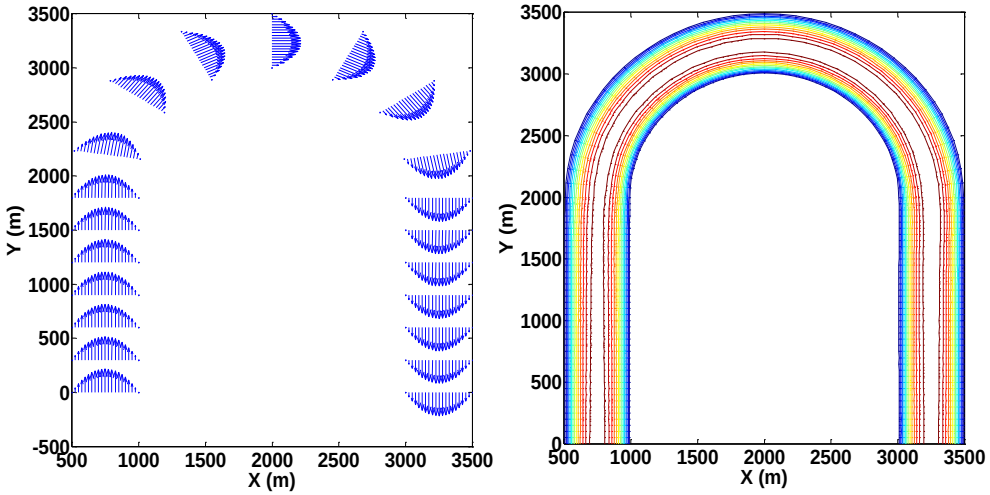


Figure 6.22 Shallow flow around a 180^o bend with inlet and outlet stems (dimensions based on idealised River Danube): bed elevation contours.

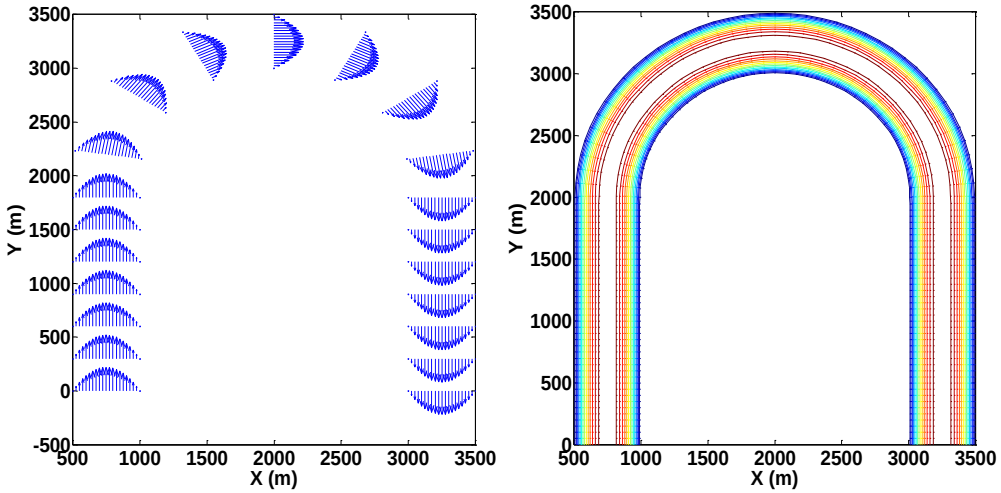


(a) (b)

Figure 6.23 Velocity time histories for shallow flow around a 180^o bend with inlet and outlet stems (dimensions based on idealised River Danube): (a) inlet v_1-velocity component; and (b) outlet v_2-velocity component time histories.



(a)



(b)

Figure 6.24 Steady-state velocity vectors and velocity magnitude contours for shallow flow around a 180^{o} bend with inlet and outlet stems (dimensions based on idealised River Danube): (a) predicted; and (b) analytical results.

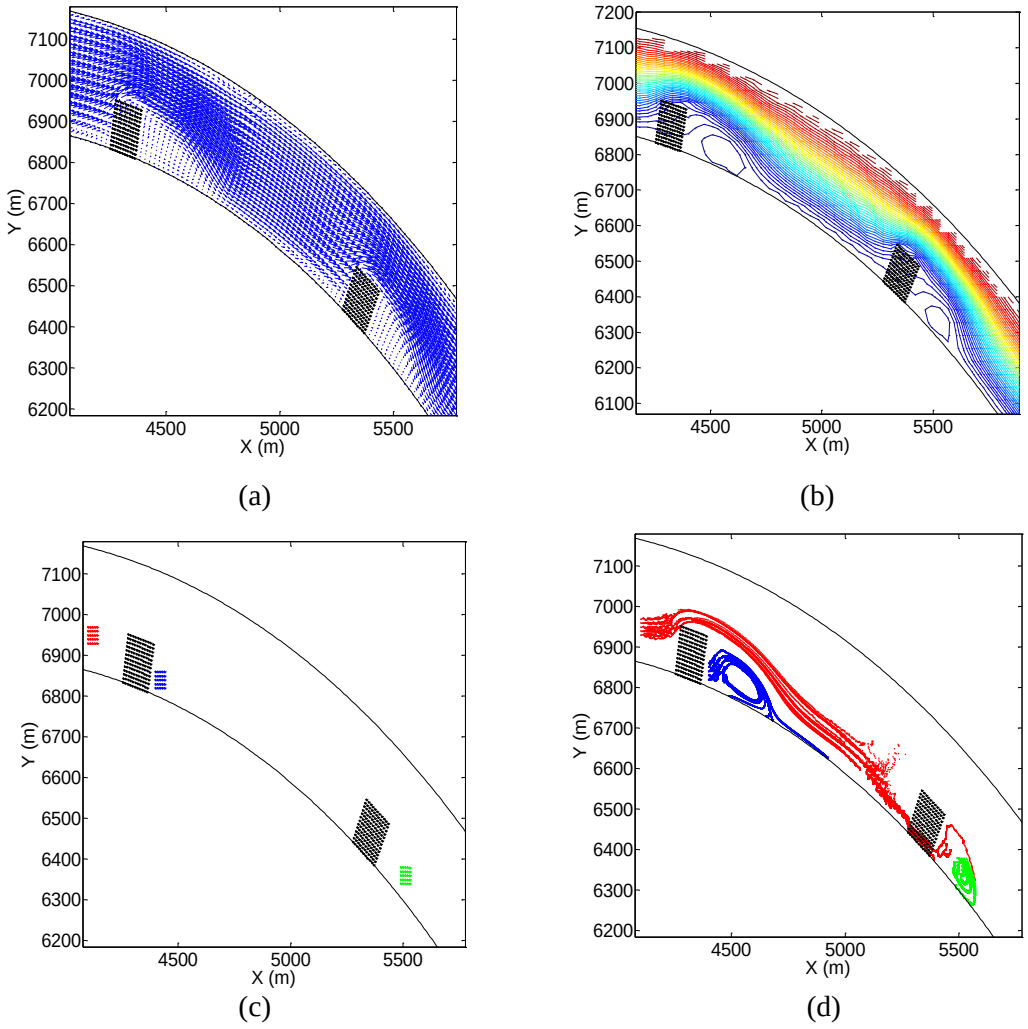


Figure 6.25 Case of a 90º river bend with two parallel groynes representative of a Danube River bend: (a) steady state velocity vectors; (b) stream function contours; (c) coloured tracer particle arrays at t = 0 s; and (d) particle positions at t = 1500 s.

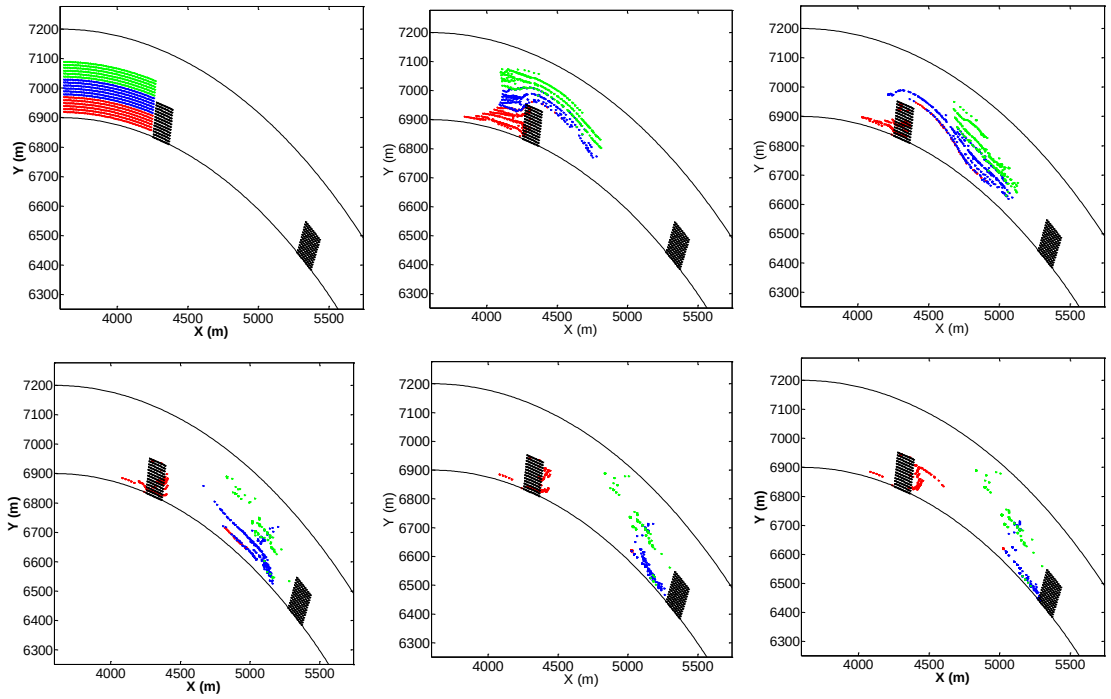


Figure 6.26 Particle advection in a 90° river bend with two parallel groynes representative of a typical Danube River bend: (a) $t = 0.0s$; (b) $t = 1000\ s$; (c) $t = 2000\ s$; (d) $t = 3000\ s$ (e) $t = 4000s$; and (f) $t = 5000s$.

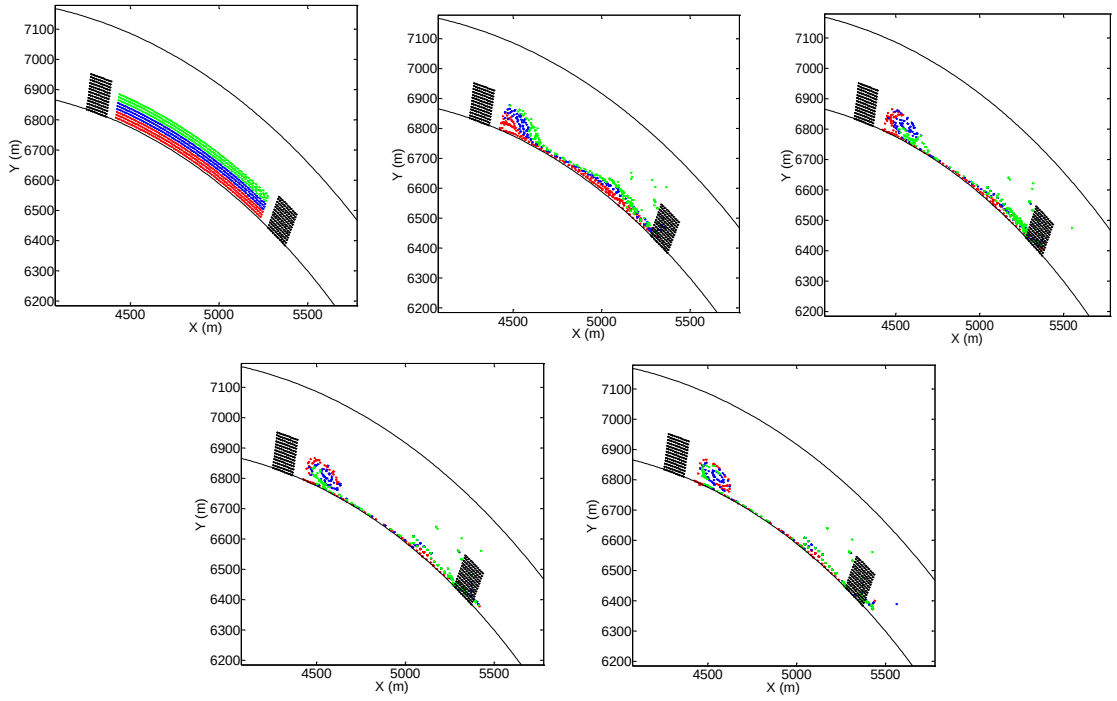


Figure 6.27 Particle advection in a 90° river bend with two parallel groynes representative of a typical Danube River bend: (a) $t = 0.0s$; (b) $t = 1000\ s$; (c) $t = 2000\ s$; (d) $t = 3000\ s$; and (e) $t = 4000s$.

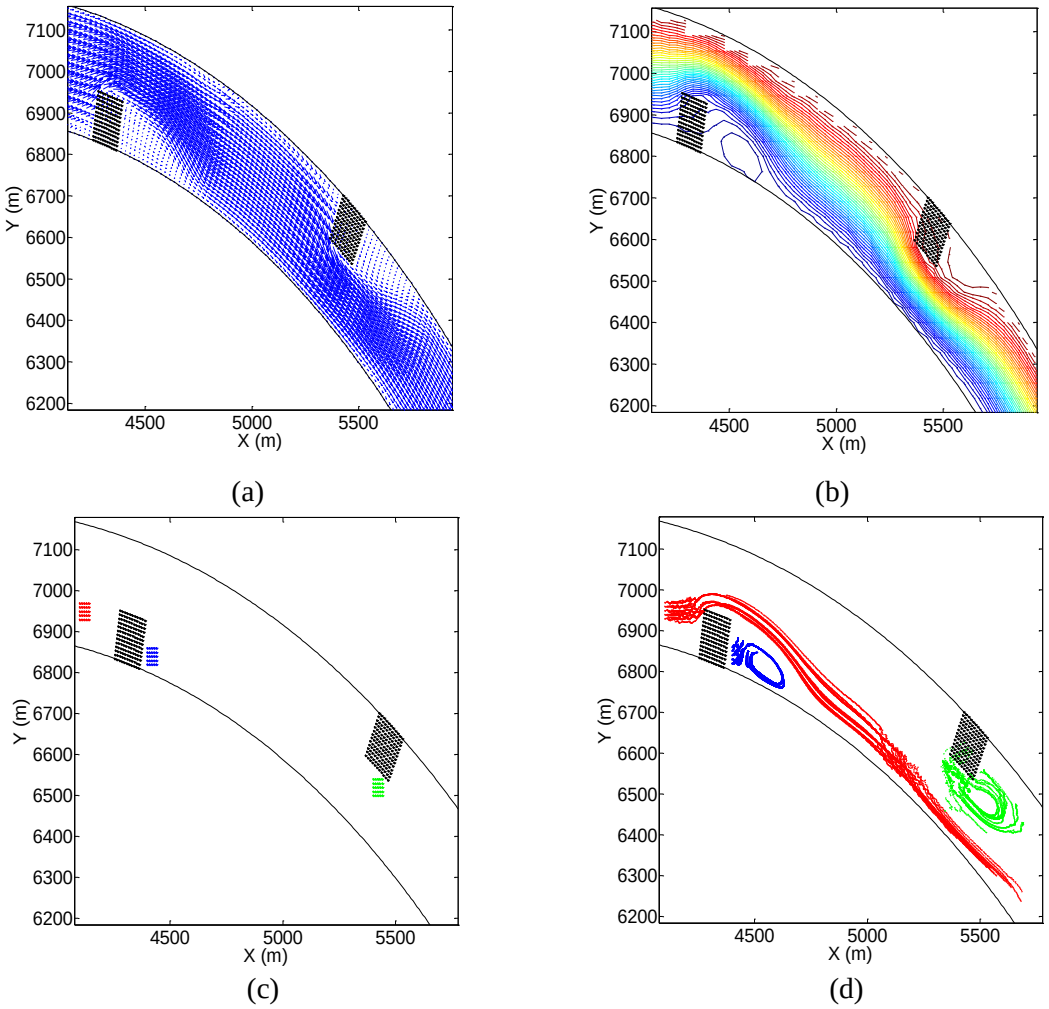


Figure 6.28 Case of a 90^{o} river bend with two alternate groynes representative of a Danube River bend: (a) steady state velocity vectors; (b) stream function contours; (c) coloured tracer particle arrays at t = 0 s; and (d) particle positions at t = 650 s.

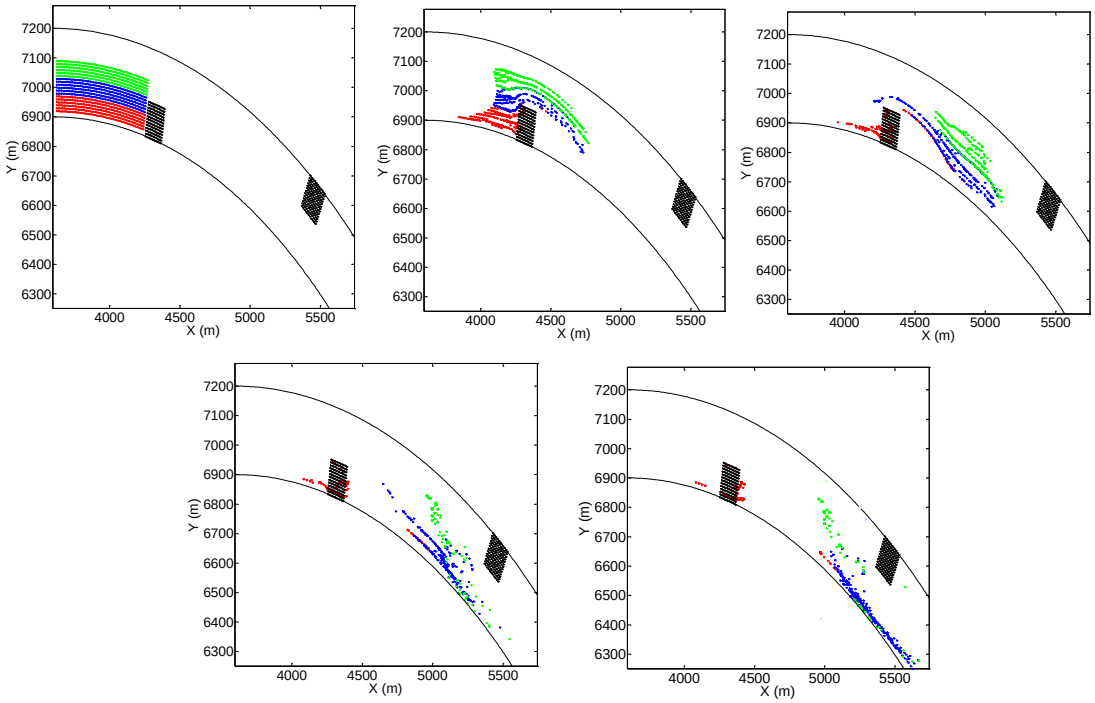


Figure 6.29 Particle advection in a 90° river bend with two alternate groynes representative of a typical Danube River bend: (a) $t = 0.0s$; (b) $t = 1000\ s$; (c) $t = 2000\ s$; (d) $t = 3000\ s$; and (e) $t = 4000s$.

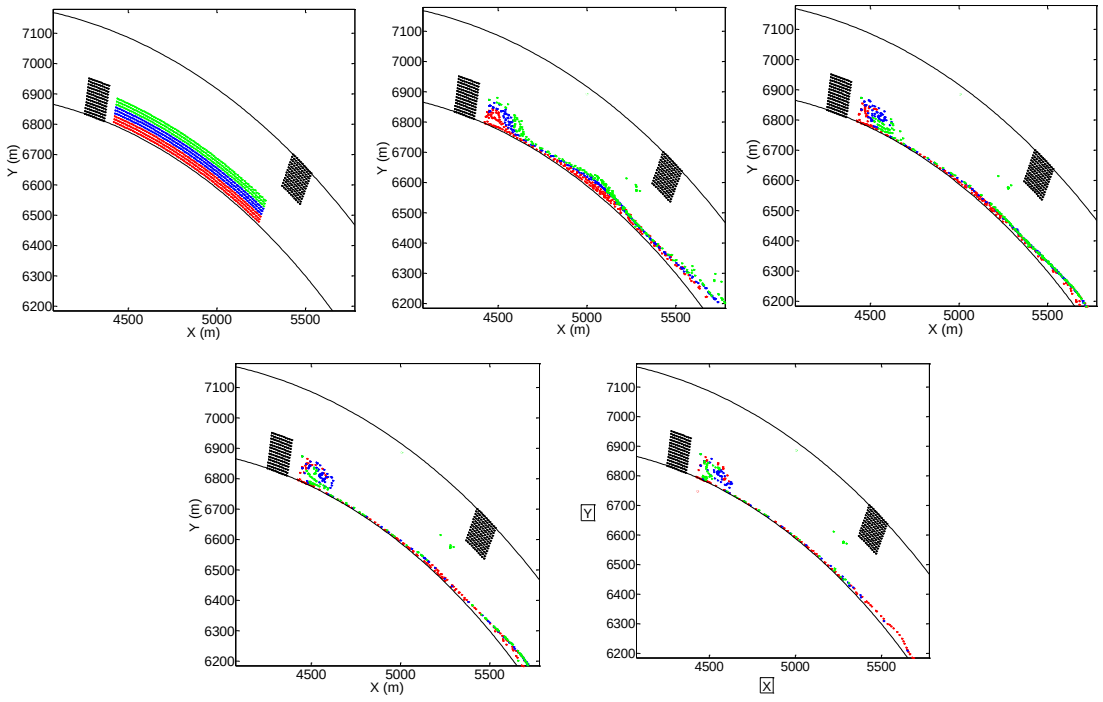


Figure 6.30 Particle advection in a 90° river bend with two alternate groynes representative of a typical Danube River bend: (a) $t = 0.0s$; (b) $t = 1000\ s$; (c) $t = 2000\ s$; (d) $t = 3000\ s$; and (e) $t = 4000s$.

Chapter 7

Conclusions and recommendations

7.1 Overall conclusions

This thesis has presented details of Cartesian and curvilinear systems shallow flow solvers which have been coupled with a Lagrangian particle tracking scheme to study advection and mixing processes in a groyned river bend. The curvilinear grid generator involved iterative solution of a pair of elliptic partial differential equations derived using Jacobian transformations and discretized using second-order central differences. Three iterative methods were considered for solving the grid generation equations: successive over-relaxation; multi-grids; and conjugate gradient, and it was found that the multi-grid and conjugate gradient techniques performed much better than the conventional successive over-relaxation method in creating curvilinear grids within a circular domain and distorted grids within a rectangular domain. A detailed derivation has been provided of a form of non-orthogonal curvilinear shallow water equations that retain the velocity components in Cartesian form. Both the Cartesian and curvilinear forms of shallow water equations have been solved using central difference approximations to the equations in space and the solution marched forward in time using second-order Adams-Bashforth and fourth-order Runge-Kutta scheme. The resulting numerical models have been verified and validated for uniform flow in a channel, wind-induced set-up in rectangular and circular basins, recirculating flow downstream of a side-wall expansion, and shallow flow in a rectangular channel with single groyne. A Lagrangian particle tracking scheme was then developed by integrating the advection equations forward in time using 2^{nd}-order Adams-Bashforth and 4^{th}-order Runge-Kutta schemes. The particle tracking algorithm was verified for a particle moving in a circular trajectory in the flow field of a free vortex, and for periodic and chaotic advection in the alternating flow field of a pair of blinkiung vortices. The coupled shallow flow and particle tracking model was then used to simulate advection and mixing processes in a channel containing a pair of side wall cavities containing a single groyne, a rectangular channel containing a pair of groynes,

and a curved channel containing three groynes oriented in parallel and alternate configurations. The latter cases are representative of the conditions at a Danube River bend, where groynes are used to train the river and help prevent flooding essentially by adding hydraulic resistance through the groynes and trapping sediment. It was observed that particles did become trapped in the recirculation zones in the lee of the groynes, depending on their initial positions. This work contained in this thesis is of relevance to the analysis of mixing processes in large rivers.

7.2 Detailed conclusions

- A non-orthogonal curvilinear grid generator has been developed based on the use of Jacobians to map a pair of Poisson equations in the Cartesian system onto a transformed system. The resulting Equations (2.14a) and (2.14b) have been discretized using second order central differences to give Equations (2.15a) and (2.15b), and these solved iteratively using three methods: successive over-relaxation, multi-grids, and conjugate gradient. The grid generator was verified for standard benchmark tests including grids within a circular domain and distorted grids within a rectangular domain. It was found that all three iterative techniques gave satisfactory results, though the performance of multi-grid and conjugate gradient methods was far superior to that of successive over-relaxation. It should be noted that the error that accumulates as a long wave across the grids in successive over-relaxation increases as grids become progressively finer, but is eliminated in both the multi-grid and conjugate gradient methods (which means that the latter exhibit better grid convergence properties).

- A detailed derivation has been given of the Cartesian shallow water equations, (3.42), (3.43) and (3.44), starting from first principles of mass and momentum conservation. Using the chain rule, the non-orthogonal curvilinear form of the shallow water equations (3.47), (3.48) and (3.49), has been derived. In this case, the velocity components are expressed in Cartesian form, following Barber (1990) Both the Cartesian and curvilinear forms of the shallow water

equations are solved after discretization in space using second-order central differences by means of an explicit time integration scheme (either 2^{nd}-order Adams-Bashforth or 4^{th}-order Runge-Kutta).

- The shallow water solvers have been verified for a series of benchmark test cases. Excellent agreement was achieved between the numerical predictions and analytical solution for uniform flow in an open channel (using both the Cartesian and curvilinear shallow flow solvers). This case tested the ability of the solvers to reconcile local acceleration, bed resistance, and bed slope terms in the shallow water equations. Excellent agreement was also obtained between the numerical predictions and analytical solutions for wind-induced set up in rectangular and circular basins, confirming that the models correctly represented local acceleration, wind stress, hydrostatic, and viscous terms in the equations (the latter used to dissipate energy). The third benchmark test comprised simulating the length of a recirculation zone downstream of a side-wall expansion in an open channel. Satisfactory agreement was achieved with results from alternative numerical schemes and previously published experimental data on air flow over a step. This test verified the correct implementation of local and convective acceleration terms and the viscous diffusion terms in the discretised governing equations. An additional test was included examining flow-structure interaction in an open rectangular channel containing a single groyne which was modelled using enhanced bed roughness; in this case, the results show that development of a recirculation zone of increasing length as the inlet Reynolds number is increased.

- A numerical particle tracking code has been developed by integrating the advection equations for a tracer using a 4^{th}-order Runge-Kutta scheme. The numerical particle tracking model was verified by following the predicted motion of a particle along a circular path under the influence of a free vortex. The particle tracking model results were in excellent agreement with the analytical solution for cases where the velocity field was determined analytically and by linear interpolation from discretised values on a grid. The

Lagrangian particle tracking model was further verified against analytical solutions of particle advection in the abruptly alternating flow field of a blinking-vortex pair. Close agreement has been obtained against the analytical results presented in the literature by Aref (1984), Khakhar and Ottino (1986) using the present particle tracking model with both a continuous velocity field and an interpolated discretized velocity field. As the stirrer strength μ increases from 0.05 to 1.5, the chaotic region grows until it fills much of the computational domain, in accordance with the findings of Aref, Khakhar and Ottino.

- The shallow water was further validated qualitatively by comparison against experimental data provided by Tuna and Rockwell (2014) for shallow flow in an open channel containing a pair of adjacent side-wall cavities. Satisfactory agreement was found between the numerical predictions by the Cartesian SWEs solver and Tuna and Rockwell findings. Results were also presented for the advection of different coloured trace particles in the region of the side-wall cavities. The results showed that the majority of particles remained trapped in the side-wall cavities, and there was a tendency for the particles to roll up into a whorl filling each cavity.

- Mixing of particles in a rectangular channel containing a pair of parallel groynes, oriented perpendicular to the lateral walls, has also been simulated using the coupled shallow flow and particle tracking model. The use of different colour bands of particles shows the trapping that occurs immediately downstream of each groyne within the recirculation zones, with the majority of particles passing over and around the groynes in the main through-flow stream.

- The combined shallow flow and particle tracking code has been used to simulate particle advection around a 90^o circular bend and a 180^o circular bend in both cases with and without inlet and outlet stems. Here, the dimensions of the river channel and bend have been selected to mimic those of a typical

Danube River bend. Again, the advection of differently coloured arrays of particles was examined, and the results show the trapping effect of recirculation zones immediately behind the groynes. This demonstration case has shown the capability of the present model to simulate advection and mixing processes in large river bends containing groynes at field scale.

7.3 Recommendations

Both the curvilinear systems shallow water solver and the Lagrangian particle tracking models were written from scratch by the author in Fortran, with Matlab used to visualize the results. Whereas the coupled model has been verified and to some extent validated, there is much further work that could be undertaken to improve the simulation of advection and mixing processes in groyned rivers.

- Better representation of locally intense flow features in shallow flows could be achieved by utilising adaptive hierarchical curvilinear grids, whereby the basic grid would be generated using the curvilinear grid equations, and a curvilinear quadtree approach used to build up the hierarchical grid. This would essentially provide a curvilinear form of the successful quadtree approaches taken by Rogers *et al*. 2001, 2003 and Liang and Borthwick 2009.

- The present numerical solver has been developed using finite differences and so is unable to model steep-fronted or discontinuous flows (such as at hydraulic jumps and bores). It is recommended that the model be rewritten using either a Godunov (shock-capturing) finite volume or a discontinuous Galerkin finite element scheme, with an approximate Riemann solver (such as HLLC (Toro, 2001)) used to deal with the fluxes at cell interfaces.

- It is recommended that the governing equations should be extended to cater for suspended sediment transport, deposition, and entrainment (following e.g. Cao *et al*. 2004) and bed load (see e.g. Huang *et al*. 2008, 2010).

- In cases where the flow depth is less shallow and the long wave assumption no longer applies, it is recommended that the equation set be extended to include wave dispersion and nonlinearity terms, such as provided by the Boussinesq-type and perhaps Green-Naghdi equations. These contain higher order and mixed (time and space) derivatives and so are more complicated to solve.

- The present model uses a very simple zero-equation turbulence model (i.e. constant eddy viscosity). The addition of a higher-order turbulence model, starting with the depth-averaged k-ε equations derived by Rodi (1984) and implemented by Barber (1990), would provide an improved picture of turbulence effects in the river. Large eddy simulation (see e.g. Rodi *et al*. 2013) is particularly recommended for its accuracy and computational performance.

- Very importantly, the present model is limited to two horizontal spatial dimensions by being depth-averaged. It is recommended that a three-dimensional free surface flow solver be applied to investigate the hydrodynamics of a groyned river. Candidate codes include OpenFOAM, TELEMAC-3D, MIKE 3, FLOW-3D, and Delft3D.

- Recent advances in numerical modelling techniques include lattice Boltzmann methods (see e.g. Zhou, 2004), smoothed particle hydrodynamics (SPH) (see e.g. SPHysics, Rogers *et al*. 2010), and high order discontinuous Galerkin methods (see e,g. Hesthaven and Warburton, 2008). These methods are highly appropriate to shallow flow modelling, and well worth investigating in the context of river flows.

- High quality laboratory and field data are vitally needed concerning advection and mixing processes in the vicinity of river groynes. New techniques in remote sensing such as Lidar and acoustic scanning offer considerable future opportunities.

References

Abbott M.B., Damsgaard A., and Rodenhuis G.S. (1973). System 21, "Jupiter" (a design system for two-dimensional nearly-horizontal flows, *J. Hydraul. Res*., 11: 1-28.

Abbott, M. B. (1979). Computational Hydraulics**,** Elements of the Theory of Free Surface Flows, *Pitman, London*.

Abbott, M. B., Basco, D. R. (1989). Computational fluid dynamics - An introduction for engineers. *Longman Scientific and Technical/John Wiley and Sons, Harlow, England/New York,* 425.

Agamemnon, A., Varonos., Bergeles George C. (2001). A multigrid method with higher-order discretization schemes. *International Journal for Numerical Methods in Fluids,* 35, 395–420.

Anastasiou, K. and Chan, C.T. (1997). Solution of the 2D shallow water equations using the finite volume method on unstructured triangular meshes. *International Journal for Numerical Methods in Fluids,* 24(11), 1225–1245.

Alcrudo, F., and García-Navarro P. (1993). A high-resolution Godunov-type scheme in finite volumes for the 2D shallow-water equations, *International Journal for Numerical Methods in Fluids*, 16, 489‑505.

Aref, H. (1984). Stirring by chaotic advection. *Journal of Fluid Mechanics,*143, 1–21.

Astrakhantsev G. P. (1971). An iterative method of solving elliptic net problems. *U.S.S.R. Comput. Math. and Math. Phys*, 11, 171-182.

Bachvalov. (1966). On the convergence of a relaxation method with natural constrains on the elliptic operator'(in russia), *USSR Comput.Math. and Math.phys.* , 6.5, 101-135.

Bai, W., Mingham, C.G, Causon, D.M., and Qian, L. (2009). Finite volume simulation of viscous free surface waves using the Cartesian cut cell approach. *International Journal for Numerical Methods in Fluids*, 63, 69–95.

Barber, R.W. (1990). Numerical modelling of jet-forced circulation in reservoirs using boundary-fitted coordinate systems, PhD Thesis, University of Salford, UK.

Barragy, E., Carey, G.F., Waiters, R. A. (1993). Application of Conjugate Gradient methods to tidal simulation. *J. Advances in Water Resources*, 16, 163-171.

Bermudez, A., and Vázquez, M.E. (1994). Upwind methods for hyperbolic conservation laws with source terms. *Comput Fluids*, 23(8), 1049–71.

Bonaventura, Luca., Rosatti, Giorgio. (2002). A cascadic conjugate gradient algorithm for mass conservative, semi-implicit discretization of the shallow water equations on

locally refined structured grids. *International Journal for Numerical Methods in Fluids*, 40, 217–230.

Borsboom,M., Genseberger,M., Hof, B., and Spee, E. (2014). Domain Decomposition in Shallow-Water Modelling for Practical Flow Applications. *Lecture Notes in Computational Science and Engineering 98, Springer International Publishing Switzerland.* 557-565.

Borthwick, A.G.L., Barber, RW. (1992). River and reservoir flow modelling using the transformed shallow water equations. *International Journal for Numerical Methods in Fluids*, 14, 1193–1217.

Borthwick, A.G.L. and Kaar, E. T. (1993). Shallow flow modelling using curvilinear depth-averaged stream function and vorticity transport equations. *International Journal for Numerical Methods in Fluids*. 17(5), 417–445.

Borthwick, A.G.L. and Akponasa, G.A. (1997). Reservoir flow prediction by contravariant shallow water equations. *ASCE J. of Hydraulic Engineering*, 123(5), 432-439.

Borthwick, A.G.L., León, S. Cruz., Józsa, J. (2001). The shallow flow equations solved on adaptive quadtree grids. *International Journal for Numerical Methods in Fluids*, 37: 691–719.

Bourban, S. E. (2013). Stratified Shallow Flow Modelling, PhD Thesis, The Open University, Affiliated Research Centre HR Wallingford, UK.

Briggs, Willian L., Henson, Van Edmon., Mccormick, Steve F. (1987). A Multigrid Tutorial' *Reference Book*, 1987; 1–100.

Brilly, M., Miklós, D., Schiller, H., Sass, J., Neppel, F., Kovács, P., Prohaska, S., Ilic, A., Petrovic, P., Mravcová, K., Holko, L., Kostka, Z., Miklánek, P., Stancíková, A., Rákóczi, L., Goda sr, L. (2010). Hydrological Processes of the Danube River Basin Perspectives from the Danubian Countries, *Springer Dordrecht Heidelberg London New York*. 1-436.

Cao Z., Pender G., Wallis S., and Carling P. (2004). Computational dam-break hydraulics over erodible sediment bed. *Journal of Hydraulic Engineering*, 130(7), 689–703.

Causon, D.M., Ingram D.M., Mingham C.G., Yang G., Pearson R.V. (2000). Calculation of shallow water flows using a Cartesian cut cell approach. *Advances in Water Resources*, 23(5), 545–562.

Causon D.M., Ingram D.M., Mingham C.G. (2001). A Cartesian cut cell method for shallow water flows with moving boundaries. *Advances in Water Resources*, 24, 899–911.

Chen, R., Hao,S., Liu, X., and Zhou, X. (2015). Applications of Shallow Water SPH Model in Mountainous Rivers. *Journal of Applied Fluid Mechanics*, 8(4), 863-870.

Chow, V.T., (1959), Open-channel hydraulics. *McGraw-Hill New York*, 1-680.

Day, T. J. & Wood, I. R. (1976). Similarity of the mean motion of fluid particles dispersing in a natural channel. *Wat. Resour*. 12, 655-666.

Deng, Z.-Q., Singh, V. P. & Bengtsson, L. (2001). 'Longitudinal dispersion coefficient in straight rivers', *J. Hydr.Engrg*. 127(11), 919–927.

Denham, M.K., Patrick, M.A. (1974). Laminar Flow over a Downstream-Facing Step in a Two-Dimensional Flow Channel, *journal of Chemical Engineering Research and Design*, 52(a), 361-367.

Dey. S. (2014). Fluvial Hydrodynamics, *Springer-Verlag* Berlin Heidelberg.529-562.

Falconer R.A. (1980). Numerical modelling of tidal circulation in harbours. *J. Waterway, Port, Coastal, Ocean Div., Proc. ASCE*, 106, 31-48.

Falconer, RA., Alstead, RD. (1990). Computational flow modelling in coastal and estuarine waters using irregular grid schemes. *SERC Report* Ref. No. GR/E65197, 1–18.

Falconer, R.A. and Mardapitta-Hadjipandeli, L. (1990). Some observations on nested modelling of flow and solute transport in harbours, *Proc. I.C.E., Part 2*, 89, 15-38.

Falconer, R.A., Lin, B., Wu, Y. and Harris, E. (2001). "DIVAST Reference Manual", Environmental Water Management Research Centre, Cardiff University.

Fedorenko. (1964). 'The speed of convergence of an iterative process' (in Russian), *USSR Comput. Math, and Math. Phys*, 4.3, 227-235.

Fletcher, C. (1998). Computational Techniques for Fluid Dynamics 1' *Reference Book. (2th edition), published Springer-Verlag Berlin Heidelberg*.

Fletcher, C. (1998). Computational Techniques for Fluid Dynamics 2' *Reference Book. (2th edition), published Springer-Verlag Berlin Heidelberg*.

García-Navarro, P., and Vázquez-Cendón, M.E. (2000). On numerical treatment of the source terms in the shallow water equations, *J. Comput. Fluids* 29, 951‑979.

George, Ken. (2007). A depth-averaged tidal numerical model using non-orthogonal curvilinear co-ordinates. *J. Ocean Dynamics (Springer-Verlag)*. 57(4), 363–374.

Gingold R.A. and Monaghan J.J. (1977). Smoothed particle hydrodynamics: theory and application to non-spherical stars, *Monthly Notices of the Royal Astronomical Society*, 181, 375-389.

Godunov, S. (1959). A Finite Difference Method for the computation of Discontinuous Solutions of the Equations of Fluid Dynamics, *Mat. Sb. (In Russia)*, 47, 357-393.

Gómez-Gesteira, M., Rogers, B.D. Dalrymple R.A. and Crespo, A.J.C. (2010). State-of-the-art of classical SPH for free-surface flows, *Journal of Hydraulic Research*, 48, 6-27.

Guinot,V. (2003). Godunov-type Schemes: An Introduction for Engineers 1st Edition. *Elsevier Science*.1-508.

Hanrahan, G. (2009). Modelling of Pollutants in Complex Environmental Systems, Volume I. *ILM Publications*.1-376.

Häuser J., Paap H.G., Eppel D. and Mueller A. (1985). Solution of shallow water equations for complex flow domains via boundary fitted coordinates, *International Journal for Numerical Methods in Fluids*, 5, 727-744.

Häuser J., Paap H.G., Eppel D. and Sengupta S. (1986). Boundary conformed coordinate systems for selected two-dimensional fluid flow problems. Part 2. Application of the BFG method, *International Journal for Numerical Methods in Fluids*, 6, 529-539.

Hestenes, M. R., Stiefel, E. (1952). Methods of Conjugate Gradients for Solving Linear Systems. *Journal of research of the national bureau of standards*, 49,409-436.

Hesthaven J.S. and Warburton T. (2008). Nodal discontinuous Galerkin Methods, *Springer*.1-502.

Hodgins S. (1977). An improved computational method for the shallow water equations based on the Leendertse (1967) finite-difference scheme, *Report STF60A-77058*, River and Harbour Laboratory, Norwegian Institute of Technology, Trondheim, Norway.

Huang, J., Borthwick, A.G.L., and Soulsby, R.L. (2008). One-dimensional modelling of fluvial bed morphodynamics. *J. of Hydraulic Research*. 46(5), 636-647.

Huang, J., Borthwick, A.G.L., and Soulsby, R.L. (2010). Adaptive Quadtree Simulation of Sediment Transport. *Proc. Inst. Civ. Engrs: Engineering and Computational Mechanics.* 163(EM2), 101-110.

Hubbard, M., and Garcia-Navarro, P. (1999). Balancing source terms and flux gradients in finite volume schemes. In *Proceedings of Godunov Methods*: *Theory and Applications*, Oxford, 18–22 October, Toro EF (ed.). *Kluwer Academic Publishers*: Amsterdam.

Hyunuk, An., and Soonyoung, Yu. (2012). Well-balanced shallow water flow simulation on quadtree cut cell grids. *J. of Advances in Water Resources*. 39, 60-70.

Jing, H.f., Li, C.G., Guo, Y.K., Zhu, L.J, and Li, Y.T. (2014). Numerical modeling of flow in continuous bends from Daliushu to Shapotou in Yellow River. *Water Science and Engineering*, 7(2), 194-207.

Johnson, BH. (1980). VAHM—a vertically averaged hydrodynamic model using boundary-fitted coordinates. *Misc. Paper HL-80-3, US Army Engineer Waterways Experiment Station Hydraulics Laboratory, Vicksburg, MI, USA.*

Johnson B.H. (1982). Numerical modelling of estuarine hydrodynamics on a boundary-fitted coordinate system, In J.F. Thompson (ed.) *Numerical Grid Generation*, 409-436.

Johnson B.H., Stein A.B. and Thompson J.F. (1982). Modelling of flow and conservative substance transport on a boundary-fitted coordinate system, *Proc. IAHR Symp. On Refined Modelling of Flows*, Vol 2, Paris, France, 509-518.

Kasvi, E., Alho, P., Lotsari, E., Wang, Y., Kukko, A., Hyyppä, H., and Hyyppä, J. (2015). Two-dimensional and three-dimensional computational models in hydrodynamic and morphodynamic reconstructions of a river bend: sensitivity and functionality. *J. Hydrological Processes*, 29, 1604–1629.

Kim, H.J., Lee, J.W., and Cho, Y.S. (2010). Numerical Simulation of Shallow-Water Flow Using a Modified Cartesian Cut-Cell Approach. 136(3). 399-404.

Khakhar, D.V., Rising, H., Ottino, J.M. (1986). Analysis of chaotic mixing in two model systems. *J. Fluid Mech*, 172,419-451.

Khakhar, D.V., Ottino, J.M. (1986). Fluid mixing (stretching) by time periodic sequences for weak flows. *Journal of Physics of Fluids (American institute of physic)*. 29 (11), 3503-3505.

Kranenburg, C. (1992). Wind-driven chaotic advection in a shallow model lake. *Journal of hydraulic research*, 30 (1), 29-46.

Kreyszig, E. (2011). Advanced Engineering Mathematics. *Reference Book (10th edition), published John Wiley & Sons*, 1-1283.

Kuipers J., and Vreugdenhil C.B. (1973). Calculations of two-dimensional horizontal flow, *Research Report S163 Part I*, Delft Hydraulics Laboratory, The Netherlands.

Lamb, H. Hydrodynamics. (1932). *Cambridge University Press*, 6th edition. 1-738.

Leendertse J.J. (1967). Aspects of a computational model for long-period water-wave propagation, *Report RM-5294-PR*, The Rand Corporation, USA.

Leveque, R.J. (1998). Balancing source terms and flux gradients in high-resolution Godunov methods: the quasi-steady wave-propagation algorithm. *J Comput Phys*, 146(1), 346–65.

Li, B., Fleming, C.A. (1997). A three dimensional multi-grid model for fully nonlinear water waves. *Journal of Coastal Engineering*, 30, 235-258.

Li, S., Huang,P., and Li, J. (2014). A modified lattice Boltzmann model for shallow water flows over complex topography. *International Journal for Numerical Methods in Fluids*, 77, 441–458.

Liang, Q., Borthwick, A.G.L., Stelling, G. (2004). Simulation of dam- and dyke-break hydrodynamics on dynamically adaptive quadtree grids. *International Journal for Numerical Methods in Fluids*, 46, 127–162.

Liang, Q., Borthwick, A.G.L., Taylor, Paul H. (2006). Wind-induced chaotic advection in shallow flow geometries. Part I: Circular basins. *Journal of Hydraulic Research*, 44(2), 170–179.

Liang, Q., Zang, J., Borthwick, A.G.L., and Taylor, P.H. (2007). Shallow flow simulation on dynamically adaptive cut-cell quadtree grids. *International Journal for Numerical Methods in Fluids*, 53(12), 1777-1799.

Liang, Q., and Borthwick A.G.L. (2008). Simple treatment of non-aligned boundaries in a Cartesian grid shallow flow model. *International Journal for Numerical Methods in Fluids*, 56(11), 2091-2110.

Liang. Q., Du, G., Hall, J.W., Borthwick, A.G.L. (2008). Flood Inundation Modeling with an Adaptive Quadtree Grid Shallow Water Equation Solver. *Journal of hydraulic engineering*, 137, 1603-1610.

Liang, Q., and Borthwick A.G.L. (2009). Adaptive quadtree simulation of shallow flows with wet-dry fronts over complex topography. *Computers & Fluids*, 38(2), 221-234.

Liang, Q. (2010). A structured but non-uniform Cartesian grid-based model for the shallow water equations. *International Journal for Numerical Methods in Fluids*, 66, 537–554.

Lin B, Falconer RA. (1995). Modelling sediment fluxes in estuarine waters using curvilinear coordinate system. Estuary. *Coastal and Shelf Science*, 14, 413–428.

Lin, B.L. and Falconer R.A. (1995). Modelling sediment fluxes in estuarine waters using a curvilinear co-ordinate grid system. *Estuarine, Coastal, and Shelf Science*, 41(4), 413-428.

Liu G.R. and Liu M.B. (2003). Smoothed particle hydrodynamics: a meshfree particle method, *World Scientific*, Singapore. 1-453.

Lucy, L.B. (1977). Numerical approach to testing the fission hypothesis, *Astronomical Journal*, 82, 1013-1024.

Lv, Sui-ju,. and Feng, Min-Quan. (2011). Three-dimensional numerical simulation of the Yellow River flow in Shapotou river bends. *Water Resource and Environmental Protection (ISWREP), International Symposium*.1, 233 – 236.

Mitchell, B.L., Fulton, R. (2000). Adaptive Multigrid Solution of the Shallow Water Equations. Technical report No 2000-02. *Department of mathematics and computer*

science clarkson university, potsdam, New York. The work was supported by naval research marine meteorology and Atmospheric effects program grants. 1-28.

Mohd-Yosuf J. (1997). Combined immersed-boundary/b-spline methods for simulations of flow in complex geometries. *Annual Research Briefs, Center of Turbulence Research* (Stanford, CA: NASA Ames Res. Center for Turb. Res.) pp 317–28.

Nash S. and Hartnett M. (2010). Nested circulation modelling of inter-tidal zones: details of a nesting approach incorporating moving boundaries, *Ocean Dynamics*, 60(6), 1479-1495.

Nash S. and Hartnett M. (2014). Development of a nested coastal circulation model: boundary error reduction. *Environmental Modelling & Software*, 53, 65-80.

Nicolaides, R. A. (1977). On the convergence of an algorithm for solving finite element equations, Math. Comp. 31**.** , 140, 892–906.

Novak, P., Moffat, A.I.B., Nalluri, C. (2007). *Hydraulic Structures*（Fourth edition), *published Taylor & Francis*. 1-700.

O'Leary, Dianne P. (2009). Some History of Conjugate Gradients and Other Krylov Subspace Methods. *SIAM Applied Linear Algebra Meeting* 2009. Computer Science Dept. and Institute for Advanced Computer Studies University of Maryland. 2009.

Ottino, J. M. (1990). Mixing chaotic advection and turbulence. *Journal of Fluid Mech,* 22**,** 207–253.

Paglieri, L., Ambrosi, D., Formaggia, L., Quarteroni, A., Scheinine, A.L. (1997). Parallel computation for shallow water flow: A domain decomposition approach. *Journal of Parallel Computing*, 23, 1261–1277.

Pearson, R.V. and Barber, R.W. (1998). Modelling depth-integrated coastal pollution using a Lagrangian particle technique. *J. Transactions on Ecology and the Environment*, 17. 233-240.

Pinka, P.G. "Danube River (river, Europe)." Available at http://www.britannica.com/EBchecked/topic/151250/Danube-River

Qiu, J. and Fang, X. (2009). Challenges on Three Dimensional Simulations of Free Surface Flow. *World Environmental and Water Resources Congress Kansas City, Missouri, United States*. 1-10.

Roache, P. J. (1998). Verification and Validation in Computational Science and Engineering. *Hermosa Publishers*, Albuquerque. 1-446.

Rodenhuis, GS. (1994). Two-dimensional nearly-horizontal flow models. In *Coastal, Estuarial and Harbour Engineers*' *Reference Book*, Abbott MB, Price WA (eds). E & FN Spon: London, 129–144.

Rogers, B.D, Fujihara, M., Borthwick, A.G.L. (2001). Adaptive Q-tree Godunov-type scheme for shallow water equations. *International Journal for Numerical Methods in Fluids*, 35, 247–280.

Rogers, B.D., Borthwick, A.G.L. and Taylor, P.H. (2003). Mathematical balancing of flux gradient and source terms prior to using Roe's approximate Riemann solver. *Journal of Computational Physics*, 192(2), 422-451.

Rogers, B.D., Dalrymple, R.A., & Stansby, P.K. (2010). Simulation of caisson breakwater movement using 2-D SPH. *Journal of Hydraulic Research*, *48*(1), 135-141.

Rodi, W. (1984). Turbulence models and their application in hydraulics- A state of the art review, 2nd Edition, *IAHR, Rotterdamseweg* 185 – P.O. Box 177, 2600 MH Delft, The Netherlands.

Rodi, W., Constantinescu G., and Stoesser, T. (2013). Large-eddy simulation in hydraulics, *IAHR Monograph, CRC Press, Taylor & Francis Group*. 1-249.

Rutherford, J. C. (1981). Hand book on mixing in rivers, *Water & soil miscellaneous publication* no. 26 1-60.

Rutherford, J. C. (1994). River Mixing. *John Wiley and Sons*, 1-347.

Sabersky R.H., Acosta A.J., Hauptmann E.G. (1989). *Fluid Flow (third edition)*. Macmillan: New York, p 230.

Schlichting, H., Gersten, K. (2000). Boundary-Layer Theory' *Reference Book. (8th edition), published Springer-Verlag Berlin Heidelberg.*1-795.

Scott, LJ., Barber, RW. (1994). The effects of grid adaptation on tidal propagation in a boundary-fitted coordinate system. In *Proceedings of the* 4*th International Conference on Numerical Grid Generation in CFD*, Weatherill NP (ed.). Pineridge Press: Swansea, 603–614.

Spitaleri, R.M., and Corinaldesi, L. (1997). A multigrid semi-implicit finite difference method for the two-dimensional shallow water equations. *International Journal for Numerical Methods in Fluids*, 25(11), 1229-1240.

Socolofsky, S.A and Jirka G.H. (2005). Special Topics in Mixing and Transport Processes in the Environment 5 th Edition. *Coastal and Ocean Engineering Division Texas A&M University M.S. 3136 College Station*, TX 77843-3136, 1-172.

Springer, S. R., Samelson, R. M., Allen, J. S., Egbert, G. D., Kurapov, A. L., Miller, R. N., and Kindle, J. C. (2009). A nested grid model of the Oregon Coastal Transition Zone: Simulations and comparisons with observations during the 2001 upwelling season. *Journal of Geophysical Research*, 114, 1-22.

Stansby, PK., Lloyd, P. (1995). A semi-implicit Lagrangian scheme for 3D shallow water flow with a two-layer turbulence model. *International Journal for Numerical Methods in Fluid*, 20, 115–133.

St-Cyr, A., Jablonowski, C., Dennis J.M., Tufo, H.M., and Thomas, S. J. (2007). A Comparison of Two Shallow Water Models with Non-Conforming Adaptive Grids: classical tests. *American Meteorological Society*.137, 1898-1922.

Stefanescu, R. and Navon, I.M. (2013). POD/DEIM nonlinear model order reduction of an ADI implicit shallow water equations model. *Journal of Computational Physics*, 237 95–114.

Thompson, J.F, Thames, F.C., Martin, C.W. (1974). Automatic numerical generation of body fitted curvilinear coordinate systems on fileds containing any number of arbitrary two dimensional bodies. *Journal of Computational Physics*, 15, 299–319.

Thompson, J.F, Thames, F.C., Martin, C.W. , Shanks Samuel P. (1975). use of numerically generated body-fitted coordinate system for solution of the Navier-Stokes equations. *AIAA second computational fluid dynamics conference*, June. Research sponsored by Langley research centre, NASA, Grant NGR 25-001-055 and office of Naval research, USN, contract NO0014-74-C-0373-P0001.

Thompson, J.F., Thames, F.C., Martin, C.W. (1977). Boundary-Fitted Curvilinear coordinate systems for solution of partial differential equation on fields contacting any number of arbitrary two dimensional bodies. *NASA contractor report sponsored by NASA Langley research centre*,NASA CR-2729:1-225.

Thompson J.F., Soni B.K. and Weatherill N.P. (Eds.) (1999). *Handbook of Grid Generation*, CRC Press, London.

Tong,C., and Meng, Y. (2006). A Numerical Shallow Water Model Based on the Non-Orthogonal Curvilinear Grids. *25th International Conference on Offshore Mechanics and Arctic Engineering Hamburg, Germany*. 4, 813-818.

Toro, E.F. (2001). Shock-capturing methods for free-surface shallow flows, *John Wiley & Sons Ltd, Chichester*. 1-309.

Triki. (2014). Multiple-grid finite element solution of the shallow water equations: water hammer phenomenon. *J. Computers & Fluids* , 90, 65–71.

Turnbull, M.S. (1995). Efficient body-fitted meshing for two dimensional flow domains. Technical report. *Department of Engineering science Jesus College University of Oxford UK*.1-79.

Tseng Y.-H. and Ferziger J.H. (2003). A ghost-cell immersed boundary method for flow in complex geometry. *Journal of Computational Physics*, 192,593–623.

Tuna, B. A., Rockwell, D. (2014). Self-sustained oscillations of shallow flow past sequential cavities. *Journal of Fluid Mechanics*, 758, 655-685.

Tung, T.Vu., Phuoc, K.T. Nguyen., Lloyd, H.C. Chua., and Adrian, W.K. Law. (2015). Two-Dimensional Hydrodynamic Modelling of Flood Inundation for a Part of the Mekong River with TELEMAC-2D. *British Journal of Environment & Climate Change*. 5(2), 162-175.

Wang, J. P., Borthwick, A. G. L., Eatock Taylor, R. (2004). Finite-volume-type VOF method on dynamically adaptive quadtree grids. *International Journal for Numerical Methods in Fluid*, 45, 485–508.

Vacondio R., Rogers B.D., Stansby P.K. and Mignosa P. (2013). Shallow water SPH for flooding with dynamic particle coalescing and splitting. *Advances in Water Resources*. 58, 10-23.

Vázquez-Cendón, M.E. (1999). Improved treatment of source terms in upwind schemes for the shallow water equations in channels with irregular geometry. *Journal of Computational Physics*, 148, 497–526.

Vanderkimpen,P., Melger, E., and Peeters. P. (2009). Flood modelling for risk evaluation – a MIKE FLOOD vs. SOBEK 1D2D benchmark study. *J. Flood Risk Management Taylor & Francis Group*, 77-84.

Weitbrecht, Volker., Uijttewaal, Wim., and Jirka, Gerhard H. (2003). 2-D Particle Tracking to Determine Transport Characteristics in Rivers with Dead Zones. *Int. Symp. on Shallow Flows, Delft*, 1-8.

White, F. M. (2003), Fluid Mechanics, *McGraw–Hill*.1-866.

Wijbenga J.H.A. (1985a). Determination of flow patterns in rivers with curvilinear coordinates, Proc. *21st IAHR Congress, Melbourne*, August 1985, reprinted as Publication No 352, Delft Hydraulics Laboratory, The Netherlands, October 1985.

Wijbenga J.H.A. (1985b). Steady depth-averaged flow calculations on curvilinear grids, Proc. *2nd Int. Conf. on the Hydraulics of Floods and Flood Control, Cambridge*, September 1985, 373-387.

Willemse, J.B.T.M., Stelling, G.S., Verboom, G.K. (1985). Solving the shallow water equations with an orthogonal coordinatetransformation. *Presented at the International Symposium on Computational Fluid Dynamics*, Tokyo [reprinted as Delft Hydraulics Communication No. 356, January 1986, Delft Hydraulics Laboratory, The Netherlands].

Wu, X.G., Shen, Y.M., Zheng, Y.H. and Yang Z.F. (2004). Numerical model of flow and pollutant transportation in non-orthogonal curvilinear coordinates. *J Environ Sci (China)*. 16(2), 293-6.

Yadlin, Y. Caughey, D.A. (1989). Diagonal Implicit Multigrid Solution of the Three-Dimensional Euler Equations.*11th International Conference on Numerical Methods in Fluid Dynamics Lecture Notes in Physics*, 323, 597-601.

Yang G., Causon D.M., Ingram D.M., Saunders R., Batten P. (1997). A cartesian cut cell method for compressible flows. Part A: static body problems. *Aeronautical Journal*, 101(1002), 47–56.

Ye T., Mittal R., Udaykumar H.S. and Shyy W. (1999). An accurate Cartesian grid method for viscous incompressible flows with complex immersed boundaries. *Journal of Computational Physics*, 156(2), 209–240.

Yost, S.A., Rao, P. (2001). A multiple grid approach for open channel flows with strong shocks. *Journal of Applied Mathematics and Computation*, 124, 381-395.

Young, D.M., Jea, K.C. (1980). Generalized Conjugate-Gradient Acceleration of Non symmetrizable Iterative Methods. *Journal of linear algebra and its applications*, 34,159-194.

Zhou, J.G. (2004). Lattice Boltzmann methods for shallow water flows, *Springer, London.*

Zhou, J.G. and Borthwick, A.G.L. (2011). Lattice Boltzmann method for variable density shallow water equations. *J. Computers & Fluids.* 49, 146–149.

Zsugyel, M., Tél, T., Józsa, J. (2014). Numerical investigation of chaotic advection past a groyne based on laboratory flow experiment. *Journal of Advances in Water Resources*, 71, 81–92.